Linköping University | Department of Computer and Information Science
Master thesis, 30 ECTS | Computer science and engineering
202019 | LIU-IDA/LITH-EX-A--2019/054--SE

Marketing Mix Modelling: A comparative study of statistical models

En jämförelsestudie av statistiska modeller i en Marketing Mix Modelling-kontext

Richard Wigren
Filip Cornell

Supervisor : Cyrille Berger
Examiner : Ahmed Rezine

LINKÖPING UNIVERSITY

Linköpings universitet
SE–581 83 Linköping
+46 13 28 10 00 , www.liu.se

Upphovsrätt

Copyright

Abstract

Deciding the optimal media advertisement spending is a complex issue that many companies today are facing. With the rise of new ways to market products, the choices can appear infinite. One methodical way to do this is to use Marketing Mix Modelling (MMM), in which statistical modelling is used to attribute sales to media spendings. However, many problems arise during the modelling. Modelling and mitigation of uncertainty, time-dependencies of sales, incorporation of expert information and interpretation of models are all issues that need to be addressed. This thesis aims to investigate the effectiveness of eight different statistical and machine learning methods in terms of prediction accuracy and certainty, each one addressing one of the previously mentioned issues. It is concluded that while Shapley Value Regression has the highest certainty in terms of coefficient estimation, it sacrifices some prediction accuracy. The overall highest performing model is the Bayesian hierarchical model, achieving both high prediction accuracy and high certainty.

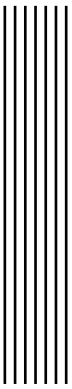
Acknowledgments

We would like to thank Goran Dizdarevic for his great commitment and contribution with expertise within the area, and for always providing a helping hand when needed. Furthermore, we would like to thank Ahmed Rezine and Cyrille Berger for the help with providing structure of the thesis. We would also like to thank the analysts at Nepa that have contributed with possible improvements and ideas regarding the experiments.

Contents

List of Figures

List of Tables



Abbrevations and explanations

ACF	Auto-correlation function
AFA	Additive Feature Attribution
AIC	Akaike's Information Criterion
AR-process	Auto-Regressive process
ARMA-process	Auto-Regressive Moving-Average process
BC	Bias corrected
BLUE	Best Linear Unbiased Estimator
BSTS	Bayesian Structural Time-Series
CART	Classification and Regression trees
CCI	Consumer Confidence Index
cdf	Cumulative distribution function
CV	Cross-validation
i.i.d.	independent and identically distributed
M	The set of media variables
MA-process	Moving-Average process
MAE	Mean Absolute Error
MAPE	Mean Absolute Percentage Error
MCMC	Markov-Chain Monte Carlo
MLE	Maximum Likelihood Estimator
MMM	Marketing Mix Modelling
OLS	Ordinary Least Squares
P	The set of predictors
$P \setminus M$	The set of control variables
PACF	Partial auto-correlation function
pdf	probability density function
RMSE	Root Mean-Squared Error
ROI	Return On Investment
RSS	Residual Sum of Squares
STL	Seasonal and Trend decomposition using LOESS
SVR	Shapley Value Regression
T	The set of time-point, $T = \{1, .., T\}$
VIF	Variance Inflation Factor

1 Introduction

'Marketing Mix' is a term first coined in 1949 by Neil H. Borden, describing a business executive as a "mixer of ingredients", creating a mix of marketing strategies to enable a profitable enterprise [7]. Borden describes the market forces as being consumers' buying behaviour, the trade's behaviour, competitors' position and governmental behaviour. Marketing Mix Modelling (MMM) is part of the Marketing Mix concept, which is a tool that allows companies to statistically model and evaluate the outcomes of their marketing efforts in different channels.

MMM is, having its response variable as sales of specific products, a regression problem, in which one attempts to model the sales as precisely as possible through a regression model, incorporating as many different variables as possible. To account for external factors and other control variables affecting sales, one includes variables such as Gross Domestic Product (GDP), precipitation and Consumer Confidence Index (CCI). A very simple additive, intuitive can be seen in Equation 1.1. Normally however, one uses significantly more complex models.

$$Sales_t = Radio_t + TV_t + Display_t + Facebook_t + Newspapers_t + Precipitation_t + GDP_t + CCI_t + \varepsilon_t \tag{1.1}$$

MMM goes under the field of demand studies within econometrics, as one predicts the demand for a certain product in one or several geographical regions over a period of time [63]. However, the main outcome of the model is not to predict the sales $\hat{y}_t$, but rather to distinguish the effects of the different marketing strategies and set these apart from other influencing factors. One can therefore consider Return On Investment (ROI) for each media spending allocation as the main outcome of a Marketing Mix Model, which should as accurately as possible say how much every dime allocated has contributed to increased sales [19, 53, 25]. The most common type of model used is a parametric regression model, in which one can infer the ROI-estimates from the parameters.

Due to above mentioned reasons, a MMM-model has three important components: its quality of fit, in other words how well its predictions $\hat{y}_t$ corresponds to the actual sales y_t, the possibility to interpret it and the confidence level of the interpretation. While the quality of fit is usually measured in generalisation error metrics such as root-mean squared error (RMSE), mean absolute error (MAE) and R^2,[1] the confidence level of the interpretation can, in a parametric regression setting be measured by examining confidence intervals on the estimates

[1] See method section for definitions

of the parameters $\mathbf{\Phi}$. Thus, one aims to achieve a model with as tight confidence intervals as possible on these parameters, to attribute sales to different media spends as confidently as possible. Further, while there exists methods to attribute sales to specific media variables in a parametric regression model, interpreting and attributing sales to specific media variables in a non-parametric setting is more difficult.

The MMM differs from a traditional regression model, such as the ordinary least squares, due to the sequential nature of the data. Data recorded in a time sequence tends to have correlated errors and not uncorrelated, which is one of the assumptions made in the ordinary least squares model [64]. Thus, a MMM-model is usually based on regression on variables experiencing effects from variables from previous time-units. This can have implications on both the complexity of the model and how the results of the model should be interpreted and optimised. Money spent on advertising a previous month might for example increase sales at a later date and an optimal model should take this into account.

Furthermore, sudden unpredictable and temporary changes of different nature can lead to one period of data differing largely from the rest. This can pose a problem for the quality of fit for a model as just a single vastly differing point of data can spoil an ordinary parametric regression model estimate [45]. Methods counteracting this, usually called robust methods, can be beneficial in such cases.

There are other big issues that have to be accounted for when creating models for MMM. Problems such as shape effects, the non-linearity of marketing expenditure, variable selection and how to incorporate expert knowledge and extracting seasonal and trend components leaves room for complex models to be developed. On the other hand, the relatively small size of MMM data sets restricts the model's complexity as it restricts the amount of parameters which can be estimated to some certainty. A number of different models have thus been tried alongside the previously mentioned models. Bayesian approaches have been used to include an informative prior [48, 19], although research including variables apart from the media coefficients is somewhat lacking. However, regression models constructed using bayesian statistics are useful in many ways. Not only can they allow for including prior knowledge and parameter dependence on multiple levels; dimensionality reduction and regularisation can also be incorporated [38].

Another issue is that of multicollinearity. Multiple marketing channel subsets might have correlated spending and therefore correlated effect on sales. While it does not affect the prediction accuracy, this has an impact on the interpretation of a model and might for example weigh importance of one explanatory variable higher than it should and another lower. There are many proposed solutions, such as Shapley value regression (c.f. [52]), and the problem warrants investigation.

An important note is that a regression model is built of correlations and not necessarily causal relations [37]. As a result, the model answers questions regarding observations rather than actions taken by using the model. While MMMs try to answer causal questions, Chan et al. [19] explain why some of the standard ways of measuring causal effects are infeasible in an MMM context and as a result regression models are used instead.

This thesis will try to investigate and compare some of the above mentioned statistical models and determine the consequences in the three important components mentioned (quality of fit, interpretation and estimation certainty) of each model in the context of two data sets; one simulated and one actual.

1.1 Motivation

Increasing computational power and larger access to data has made it possible to use new models and methods to approach the marketing mix concept. It is in the interest of organizations to optimize their spending for the expected or wanted result, which raises interest in both informative and accurate models. So far, there exists a lack of research and investigation

which compares several models and their properties, examining the differences and determining which aspects of MMM are the most important. It is therefore relevant to compare models with different properties to identify what might be the most important aspects in the context of MMM.

1.2 Aim

The underlying goal of this thesis is to explore marketing mix models by investigating models handling problems within MMM and examine the effects and outcomes of these models. More specifically, to determine what model might be the most suitable to use in a MMM context with regards to the quality of fit, the uncertainty of the model and its interpretation. This is done in collaboration with Nepa, a market research company located in Stockholm.

1.3 Research questions

Within the context of MMM and one simulated and one real dataset provided by Nepa, the following questions will be considered.

1. Can models that aims to mitigate uncertainty improve the quality of fit and tightness of the confidence intervals of the parameters and ROI estimates in a MMM context?

   In classical regression models, different types of uncertainty can be handled through robust regression or Shapley Value Regression, but there is a lack of research as to how to apply these in an MMM context. It is therefore of interest how effective these are and whether these should be used in MMM.

2. Can modelling time-dependency improve the quality of fit and tightness of the confidence intervals on the parameters?

3. What effect will incorporating prior information into MMM model have in terms of quality of fit and tightness of the confidence intervals on the parameters?

   In Bayesian models, prior information can be incorporated through the use of prior distributions, while boundaries and constraints on parameters might be used in classical regression models. However, what effect this has can widely depend on the structure of the data and the context, and is thus of interest to investigate.

4. Can a non-parametric regression model be used to obtain ROI-estimates and at what certainty?

1.4 Delimitations

Despite relying on causal effects to properly answer the advertisers questions, attempts to identify causal relations will not be made. The issue of causality will however be considered when relevant, and results will be interpreted thereafter. Secondly, while choice of method of extracting seasonal and trend components is an important issue to address, this will not be covered by this thesis. Further, despite variable selection being an important issue within MMM, this topic will not be researched and covered. Lastly, while the thesis tries to explore marketing mix models in a broader sense, the evaluation of models are limited to one simulated and one real dataset.

2 Theory

2.1 Regression

As seen in Definition 9, Marketing Mix Modelling is regarded as a regression problem [19, 25] used to describe the relations between the sales and other effects, such as marketing spend. This in turn enables prediction of future sales. The sales can in such a setting be called the *response variable*, which will be denoted by Y. The model then includes other variables which might help explain the behaviour of the response variable. A common name for these variables is *predictors* and they will be denoted by $\mathbf{X} = X_1, X_2, ..., X_p$.

In a general context, the most common regression modelling tries to describe how the mean of the response variable changes with changing conditions, while the variance remains unchanged [64]. This can be described by the regression function which reads

$$\mathbb{E}[Y|\mathbf{X}] = g(\mathbf{X}). \tag{2.1}$$

While the goal of finding $g(\mathbf{X})$ is the most common, benefits can be derived from extending the modelling. Modelling the distribution of the response variable given the predictors $f(y|\mathbf{X})$ allows for the modelling to express uncertainty of the response $\mathbf{y}$ [6].

2.1.1 Model uncertainty

The mean is often assumed to have a deterministic function, and instead the uncertainty within the model comes from the error term, denoted by ε. This error can be considered a "catch-all" expression, stating that the model cannot explain the behaviour of the response variable completely [37].

The relationship between the error term and response variable can take multiple forms and can have differing interactions with the regression function. Two common structures are additive and multiplicative errors. The additive error model reads

$$Y = g(X) + \varepsilon, \tag{2.2}$$

and is often a useful approximation to the truth [44]. The multiplicative error model instead has the relation

$$Y = g(X) \cdot \varepsilon. \tag{2.3}$$

and is useful for studying problems with increasing variance as the response variable increases. Although these are commonly assumed structures, the error could be affecting the response variable in any structure $Y = h(X, \varepsilon)$ which can introduce further modelling complexity.

2.1.2 Notation

So far the regression relationship has been described with random variables. However, in reality, the methods will be applied to samples of the joint distribution $f(x, y)$ of the random variables X and Y. These samples are also called observations or realizations of the belonging random variables. For simplicity, the same notation will in some cases be used for random variables and realized values. Mainly, for a set of n samples $\{(\mathbf{x}_i, y_i) : i = 1, ..., n\}$ from the joint distribution $f(x, y)$ the following notation will often be used.

$$\mathbf{X} = \begin{bmatrix} — & \mathbf{x}_1 & — \\ — & \mathbf{x}_2 & — \\ & \vdots & \\ — & \mathbf{x}_n & — \end{bmatrix}, \quad \mathbf{x}_i^T = \begin{bmatrix} x_{i,1} \\ x_{i,2} \\ \vdots \\ x_{i,n} \end{bmatrix}, \quad \mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}, \quad \varepsilon = \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_n \end{bmatrix}$$

Especially note the notation where a random variable is distributed after a probability distribution $y \sim f(y|X)$, in which case the random variable and the realized variable will both be denoted by y.

2.1.3 Assumptions of a regression model

There are many assumptions that have to be fulfilled in order to satisfy theoretical guarantees within different models. However, the following two assumptions must hold true through all models explored in future sections.

1. **The data is a random sample from the population under investigation.**
   In other words, the data is useful for gaining the information sought for. Without this fundamental assumption modelling of the relations cannot be done; the whole analysis is worthless from the start.
2. **The relationship between explanatory and response variable modeled is correct.**
   The response variable can be predicted by the assumed relation, however, it does not have to be the true underlying structure. For example, a true underlying structure

   $$Y = \frac{1}{2}(X_1 + X_2) + \varepsilon, \; Cor(X_1, X_2) = 1, \; Var(X_1) = Var(X_2)$$

   can be described adequately by both $Y = X_1 + \varepsilon$ and $Y = X_2 + \varepsilon$, but neither describe the correct causal relation. However, the relationship given above must be assumed to be correct, although it might not correctly reflect the true underlying structure.

While these assumptions can be stated in many ways, Freund et. al. [37] uses the assumption "The model adequately describes the behaviour the data" which can be seen as a combination of both.

2.1.4 Limitations of regression models

While regression methods are powerful modelling tools, there are limitations to them as well, with the causal and extrapolation limitations perhaps being the most important. First, even if all assumptions are fulfilled, a regression relationship does not imply a causal relationship

[37]. More specifically, a regression relationship between some predictor X_i and the response variable Y does not imply that X_i causes Y or vice versa. Secondly, a model fit with observed values should not be used to make inference on values outside of the observed space [37]; also known as extrapolating. Non-linear models can for example show a close to linear behavior in a subspace while containing far from a linear behavior on the whole space.

2.1.5 Estimators in regression

In some cases the task of estimating a parameter is an easy task. As an example, the sample mean is a good estimator of the population mean. However, in more complex cases there is a need for a more methodical way [18]. These methods can differ depending on the assumed structure of the estimator. There are for example both non-parametric estimators which do not estimate parameters and parametric estimators which do. Non-parametric methods, however, suffer from the curse of dimensionality which causes their expected error to raise quickly with the number of dimensions (the number of predictors) [73].

In parametric regression there is commonly a set of data $\mathcal{D} = \{(y_1, \mathbf{x}_1), ..., (y_n, \mathbf{x}_n)\}$ which is used to estimate the parameters [44]. This estimation can take various forms. Casella and Berger [18] state that the maximum likelihood estimator is by far the most frequently used while Hastie et al. [44] mention that the least squares estimator is the most common in a regression setting. These can, under some assumptions, be shown to be equivalent for regression.

2.1.5.1 Least squares estimator

In the parametric case, the least squares estimator is identified through minimising the residual sum of squares, stated as

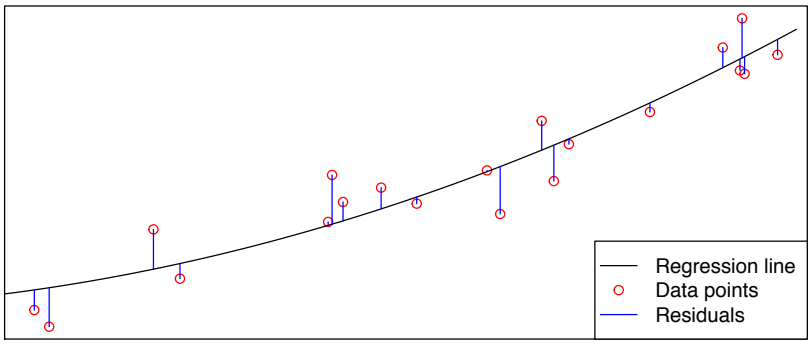


Figure 2.1: Visualisation of a quadratic regression line fit minimizing least squares and its belonging residuals.

$$RSS(f) = \sum_{i=1}^{n} (y_i - f(\mathbf{x}_i; \beta))^2 \tag{2.4}$$

with respect to the parameters β. Here, a residual refers to the difference between an estimate and its respective response variable's true value. The key idea is to minimize the difference between the true value and the predicted value. A regression line fit according to least squares with belonging residuals can be seen in Figure 2.1. Specifically, the formulation of this estimator reads

$$\hat{\beta} = \arg\min_{\beta} RSS(f). \tag{2.5}$$

There are multiple ways in which this can be optimized. In some cases, such as linear regression, there exists a closed form solution, whereas in other cases the estimator is found through other

methods such as iterative optimization algorithms. One example of this is the Gauss-Newton method, later introduced in Section 2.3.2.1.

2.1.5.2 Maximum Likelihood

The maximum likelihood estimator is found through maximising the likelihood function. The approach considers the parameters that maximise the probability of the observed sample occurring as the most reasonable [44]. A definition of the likelihood function from Casella and Berger [18] with a slightly different notation can be seen in Definition 1.

Definition 1 (Likelihood function, Casella and Berger)**.** *Let $f(y|\beta)$ be the joint probability density function or a probability mass function of the sample $Y = (Y_1, ..., Y_n)$. Then, given that $Y = y$ is observed, the function of β defined by*

$$\mathcal{L}(\beta; y) = f(y|\beta)$$

is called the ***likelihood function****.*

When f is a probability mass function, the likelihood can simply be interpreted as the probability of observing the realized data given the parameters β. The continuous case is not as trivial, but can be estimated by the same method (see Appendix A.1). In a regression context, the pdf of the response variable includes the predictors and the notation of the likelihood function is extended as

$$\mathcal{L}(\beta; X, y) = f(y|X, \beta) \tag{2.6}$$

The maximum likelihood estimator (MLE) is then found through maximizing this function with respect to the parameters, β.

2.2 Linear Regression

Linear, parametric regression models have many benefits but also limitations. If the true underlying structure is approximately linear in the predictors, then a linear model of such a problem is naturally suitable. This is of course not always the case, but they often provide an interpretable and reasonably adequate explanation of the response variable and predictors, while still remaining simple. The simplicity and interpretability are two of the greatest benefits of the linear model [44, 41].

In linear regression, the regression function is assumed to be linear in its explanatory variables $\mathbf{X} = X_1, X_2, ..., X_p$ [44]. More specifically, the model of the response Y is stated as

$$Y = \beta_0 + \sum_{i=1}^{p} \beta_i X_i + \varepsilon \tag{2.7}$$

The term β_0 is called the *intercept* and helps adjust for the mean of Y. Given a realized set of response variables and predictors and by including a constant $X_0 = 1$ in X, this can be rewritten in matrix notation

$$y = \mathbf{X}\beta + \varepsilon, \tag{2.8}$$

with $x_i = (1, x_{i,1}, ..., x_{i,p})$ and $\beta = (\beta_0, \beta_1, ..., \beta_p)^T$.

2.2.1 Assumptions

Along the general assumptions for regression models, there are a few assumptions of the linear model under which theoretical guarantees such as the Gauss-Markov theorem, explained in the next section, holds.

1. Uncorrelated errors.

The error terms, ε_i, are uncorrelated. More specifically

$$Cov(\varepsilon_i, \varepsilon_j) = 0, \quad \forall i, j : \ j \neq i \tag{2.9}$$

2. **Homoscedasticity of errors.**
   The error terms have the same variance,

$$Var(\varepsilon_i) = \sigma^2, \quad \forall i \tag{2.10}$$

3. **Exact covariates.**
   The covariates (the predictors) x_i are all known without error. While the covariates can still follow a probability distribution, the measurements of x_i must be exact.
4. **Normally distributed errors.**
   The error terms, $\varepsilon_i : i = 1, ..., n$, follow a joint normal distribution.

Freund and Wilson [37] summarize these assumptions with the statement: *"The random error is an independently and normally distributed random variable with mean zero and variance* σ^2*"*. They have been separately emphasized here but will further take the form

$$y|\mathbf{X} \sim \mathcal{N}(X\beta, \sigma^2 \mathbb{I}_n) \tag{2.11}$$

when models are discussed.

2.2.2 Estimators

Given the assumed linear model, the goal is to find an estimator of the response variable $\hat{y}(x)$ from the observed data. Since the model is parametric, this is done through the estimation of the parameters. One way to do this estimation is the aforementioned least squares approach. In the linear context, the least squares is often called Ordinary Least Squares (OLS) and the estimator reads

$$\hat{\beta}^{ols} = \arg\min_{\beta} \sum_{i=1}^{n} (y_i - \mathbf{x}_i^T \beta)^2 = \arg\min_{\beta} \|y - \mathbf{X}\beta\|_2^2 \tag{2.12}$$

Given that a fit can not be better than a perfect fit, $RSS(\hat{y}) = 0$, there has to exist a minimum larger or equal to the perfect fit. The estimator therefore has the closed form solution

$$\begin{aligned} \frac{\partial}{\partial \beta} \|y - \mathbf{X}\beta\|_2^2 = \ & 2\mathbf{X}^T(y - \mathbf{X}\beta) \stackrel{!}{=} 0 \\ \Leftrightarrow \ & \mathbf{X}^T y = \mathbf{X}^T \mathbf{X}\beta \\ \Leftrightarrow \ & \beta = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T y \end{aligned}$$

if $\mathbf{X}^T\mathbf{X}$ is non-singular. In the case of linear regression with normally distributed errors, the maximum likelihood is equivalent to the OLS (see Appendix A.2).

The Gauss-Markov Theorem states that the variance of the OLS-estimator is the smallest among all linear unbiased estimators of β [44]. Note that the OLS-estimator of β is a linear combination $c_0^T y$ of the response, $a^T\hat{\beta} = a^T\mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T y$, since $\mathbf{X}$ is considered fixed. The Gauss-Markov theorem can then be stated as in Theorem 1.

Theorem 1 (Gauss-Markov)**.** *Let*

$$Y = X\beta + \varepsilon, \quad \mathbb{E}[\varepsilon] = 0, \quad Cov(\varepsilon) = \sigma^2 \mathbb{I}_n, \quad Rank(X) = p$$

Then the least squares estimator $\hat{\beta}$ *is the best linear unbiased estimator (BLUE) of* $a^T\beta$*. More specifically:*

$$Var(a^T\hat{\beta}) \leq Var(c^T y), \quad \forall a, \ \mathbb{E}[c^T y] = a^T\beta,$$

Proof. See Halliwell [40]. □

However, this theorem does not provide a full argument for using the OLS-estimator. This is since although the OLS-estimator is the BLUE estimator, restriction to unbiased estimates might not be the best choice [44]. In accordance with the bias-variance trade-off it is possible to get more accurate predictions with biased estimators. Introduced bias in different forms is explored in later sections.

2.2.3 Violations of assumptions and remedies

While violations of the aforementioned assumptions does not prevent the use of the ordinary least squares estimator, each violation has consequences for the properties of the estimator. If the model is not correctly specified it is affected by specification error. The most common cause for this error is missing parameters or predictors and leads the coefficient and variance of the estimators to become biased [37].

1. **Uncorrelated errors.**
   When the errors are correlated, the OLS estimators will still be unbiased, but no longer the best estimators [64]; they lose the BLUE property. If specific structures exists within the correlation there are methods which can prevent this. These will be explored in Section 2.9.
2. **Homoscedasticity of errors.**
   Errors are heteroscedastic when $Var(\varepsilon_i) = \sigma_i^2 \neq Var(\varepsilon)$ for all i, j. Freund and Wilson [37] state that the variance and skewness will often be tied to the mean of the response variable as a results of heteroscedastic errors. To prevent this, they suggest the method of weighted least squares to be used when the variance changes in a systemic fashion.
3. **Exact covariates**
   If the measurement of covariates follow a distribution, for example $X_i = Z_i + \epsilon_i$ with the resulting model $Y = Z^T\beta + \varepsilon$, the covariates are not considered exact. In this case the OLS-estimator is inconsistent [18]; it does not converge to the true value with increasing number of observations. This causes a systemic error.
4. **Normality of errors**
   By the previously covered Gauss-Markov theorem (see Theorem 1), the least squares estimators will still be the best linear unbiased estimator, but makes it harder to construct confidence intervals [64].

2.3 Non-linear Regression

While the linear class of regression has many benefits, such as the simplicity, it can not reliably be used for all problems. If the true underlying structure of the relation between the response and predictors is not linear, the model might not be adequately described by a linear model. Non-linear models are instead often chosen because they are to some extent more realistic in many contexts [64]. For example, sales can be argued to have diminishing returns with increased advertisement and a model that captures this might better represent reality. These non-linear models can be split up in two categories: Intrinsically linear models that can be linearized through transformations, and those that can not [64].

2.3.1 Intrinsically linear models

Intrinsically linear models are non-linear models that become linear under transformation. An example of this is the multiplicative model. This model is stated as

$$Y = \beta_0 X_1^{\beta_1} X_2^{\beta_2} \cdots X_p^{\beta_p} \varepsilon \tag{2.13}$$

and can be made linear through a logarithmic transformation as

$$\log Y = \log \beta_0 + \beta_1 \log X_1 + \beta_2 \log X_2 + ... + \beta_p \log X_p + \log \varepsilon \tag{2.14}$$

which is linear in the parameters. The advantage of this is the ability to use ordinary least squares in this transformed space. We define an intrinsically linear model as following.

Definition 2 (Intrinsically linear models)**.** *Let $Y, X = (X_1, X_2, ..., X_p)$ and ε be random variable and assume a relation $Y = f(X, \varepsilon)$. Then, if there exists a function h such that*

$$h(Y) = h(f(X, \varepsilon)) = \sum_{j=1}^{k} \phi_j(X)\beta_j + \varepsilon, \quad \beta_j \in \mathbb{R}$$

for some $\phi_j : \mathbb{R}^p \to \mathbb{R}$ and k, the relation is called intrinsically linear.

We note that while this incorporate relations such as the multiplicative model, models with a relation

$$f(X, \varepsilon) = \sum_{j=1}^{p} \phi_j(X)\beta_j + \varepsilon$$

are also intrinsically linear through $h(y) = y$. As a result models that perform transformations on the predictors can also be intrinsically linear. This is very useful and significantly increases the scope of the linear models [44]. An example of this is the one-predictor polynomial model

$$Y = \beta_0 + X_1\beta_1 + X_1^2\beta_2 + ... + X_1^p\beta_p + \varepsilon,$$

which is linear in the coefficients.

2.3.2 Intrinsically non-linear models

The intrinsically non-linear models are instead models which can not be transformed into a linear relationship. While these can further increase the scope of regression models, they increase complexity in estimation. This estimation is commonly done through an iterative numerical method [37], but the non-linearity might cause the optimisation problem to be non-convex. In this study, intrinsically non-linear models are not considered, but optimisation methods used for these are as some of these allow to set restrictions on coefficients. There are many different optimisation methods allowing this; one being the Gauss-Newton method.

2.3.2.1 Gauss-Newton

The Gauss-Newton method is a gradient descent method, used to find optimal solutions for non-linear functions. Given a set $\mathcal{F} = \{f_1, ...f_n\}$ of functions, the objective is to minimize

$$min_{\beta \in \mathbb{R}^p} \mathcal{F}(\beta) = \sum_{i=1}^{n} f_i^2(\beta)$$

The exact method can be found in Algorithm 1 as described by Hansen et al. [22]. Important to note is that given linearly independent columns, the solution will be given by equation 2.15.

$$\beta^{(t)} = \beta^{(t-1)} - \left(\nabla\mathcal{F}(\mathbf{X}; \beta^{(t-1)})^T \nabla\mathcal{F}(\mathbf{X}; \beta^{(t-1)})\right)^{-1} \nabla\mathcal{F}(\mathbf{X}; \beta^{(t-1)})^T \mathcal{F}(\beta^{(t-1)}) \tag{2.15}$$

In a (intrinsically) linear regression context, the functions $\mathcal{F}$ can be considered to be the RSS function given in 2.4 for each data point, i.e.,

$$f_i = r_i = (y_i - F(\mathbf{X}; \beta)) \quad \forall i \in \{1, .., n\}$$

Algorithm 1 The Gauss-Newton method.

procedure GAUSS-NEWTON
  Given functions $\mathcal{F} = \{f_1, ...f_n\}$ and start-values $\beta^{(0)} = (\beta_1^{(0)}, .., \beta_p^{(0)})$
  for t in 1..T **do**
    Solve $\beta^{(t)} = \arg\min_\beta \|\mathcal{F}(\mathbf{X}; \beta^{(t-1)}) + \nabla\mathcal{F}(\mathbf{X}; \beta^{(t-1)}) \cdot (\beta - \beta^{(t-1)})\|^2$
  end for
  Return $\beta = \beta^{(T)}$
end procedure

where $F(\mathbf{X}; \beta)$ is the parametric function used to estimate $\hat{y}_i$. The hessian can therefore then be given by

$$\nabla\mathcal{F}(\beta^{(t-1)})^T \nabla\mathcal{F}(\beta^{(t-1)}) = \frac{\partial^2}{(\partial\beta)^2}\|\mathbf{X}\beta - \mathbf{y}\|^2 = \mathbf{X}^T\mathbf{X} \tag{2.16}$$

The Gauss-Newton method is suitable when the problem is mildly non-linear. There are also a few things to consider when using the Gauss-Newton algorithm. First of all, an appropriate set of starting values should be used for convergence. Secondly, if the hessian $\nabla\mathcal{F}(\beta^{(t-1)})^T \nabla\mathcal{F}(\beta^{(t-1)})$ is not of full rank, the algorithm will have a problem with convergence, as this is an assumption for the algorithm to converge [72]. In the case of (intrinsically) linear regression, a solution will therefore be difficult to find if the data matrix $\mathbf{X}$ is (nearly) singular, as this makes the hessian $\mathbf{X}^T\mathbf{X}$ (nearly) singular.

Another assumption is that a local minima or maxima exists, i.e., that there exists a β^* such that

$$\nabla\mathcal{F}(\beta^*) = \mathbf{0} \tag{2.17}$$

If there is no point such as above, the algorithm cannot converge to a single point [72]. For this assumption to hold in a linear regression context, it is required that $\mathbf{X}^T\mathbf{X}$ is positive semi-definite, which always holds as proven as below.

Theorem 2. *Let $\mathbf{X} \in \mathbb{R}^{m\times n}$, then $\mathbf{X}^T\mathbf{X}$ is always positive semi-definite.*

Proof. Let $z \in \mathbb{R}^n$ be an arbitrary vector. $\mathbf{X}^T\mathbf{X}$ is then always positive semi-definite, since it holds that

$$z^T(\mathbf{X}^T\mathbf{X})z = (\mathbf{X}z)^T(\mathbf{X}z) = \|\mathbf{X}z\|_2^2 \geq 0$$

□

It is however possible that there may be several points in a hyperplane fulfils the criteria given in Equation 2.17; in particular if the first condition described above is not met.

2.4 Bayesian regression

Bayesian statistics is a field within statistics, with its foundation in the belief in the Bayesian interpretation of probability. Inference is performed by modelling parameters' probable values through probability distributions by combining a conditional distribution, derived from the likelihood of parameters given the data and a prior belief distribution [38]. Being based on the Bayesian view on probability, almost all modelling is based on the Bayesian formula, seen in Definition 3. While the conditional distribution, conditional on the data, allows the model to adjust to the data, the prior distribution enables the possibility to incorporate a subjective belief of the true value of the parameters, thus indirectly being able to incorporate knowledge or beliefs not seen in the data.

Definition 3. *Posterior distribution*
Let $\mathbf{y} = \{y_1, .., y_n\}$ *be a collection of data with an assumed relation the parameter* θ*, which in turn is assumed to have a prior distribution* $p(\theta)$*. The posterior distribution of the parameter* θ *is then*

$$p(\theta|\mathbf{y}) = \frac{p(\mathbf{y}|\theta)p(\theta)}{p(y)} \propto p(\mathbf{y}|\theta)p(\theta) = \Big(\prod_{i=1}^{n} p(y_i|\theta)\Big)p(\theta)$$

As the normalisation often is a constant intractable to compute for many distribution, it suffices to say that the posterior is proportional ($\propto$) to the factors depending on the parameters.

An example is of a Bayesian regression model is the bayesian version of the most classical regression model, the OLS presented previously (see Section 2.2.2), and can be described by Equation 2.18.

$$y_i = \mathbb{E}(y_i|\beta, \mathbf{X}) + \varepsilon = \mathbf{X}\beta + \varepsilon, \quad \varepsilon \sim \mathcal{N}(0, \sigma^2) \tag{2.18}$$

This forms the posterior distribution

$$p(\beta, \sigma^2|y, \mathbf{X}) \propto p(\beta|\sigma^2, y, \mathbf{X})p(\sigma^2, \beta) \tag{2.19}$$

if one assumes that σ^2 and β are independent. This allows to incorporate prior beliefs into the model with the use of the prior distribution $p(\sigma^2, \beta)$, which can be formed either as a joint distribution or as two independent distributions, $p(\sigma^2)$ and $p(\beta)$.

2.4.1 Predictive inference

In Bayesian statistics, one uses distributions to derive probabilities of predictions. Often, it can be more important to obtain knowledge of the distribution, rather than obtaining an actual value. These are known as prediction distributions, and can be derived as

$$p(y_{t+1}|y_{1:t}) = \int_\theta p(y_{t+1}|\theta)p(\theta|y_{1:t})d\theta \tag{2.20}$$

where $y_{1:t}$ denotes the data given previously and θ is one or several parameters the model is dependent on. Thus, one does not necessarily receive a direct prediction out of a Bayesian model, but rather a posterior predictive distribution. While the posterior helps us derive insights on the probabilities of the parameters, it does only this; it does not provide a point estimate. From the posterior distribution, one can use a point estimate to summarise the distribution into a definite prediction.

2.4.1.1 Summarizing posterior inference

To build a model that optimises the fit of the model to the data, a loss function must be defined, used to evaluate each prediction's fit. One always wishes to minimise the expected loss, which depends on the parameters set in the model.

Definition 4. *Bayesian posterior loss*
Let $\mathcal{D} = (\mathbf{X}, \mathbf{y})_{1:T}$ *be the data,* θ *be the vector of parameters to be estimated,* δ_x *be the point estimate chosen and* $\mathcal{L}(\theta, \delta_x)$ *be the loss function defined. The expected posterior loss can then be defined as*

$$\mathbb{E}\big[\mathcal{L}(\theta, \delta_x)\big|x\big] = \int_\theta L(\theta, \delta_x)p(\theta|x)d\theta$$

The loss function differs from problem to problem. Depending on which loss function one has, on should use different point estimators. For squared loss functions, the posterior mean (Equation 2.21) is preferred, while the posterior mode is used for 0-1 loss functions, often used for classification problems. The posterior median is convenient for linear loss functions [38].

$$\delta_{Posteriormean} = \mathbb{E}[\theta|\mathbf{X}] = \int_\theta \theta \cdot p(\theta|\mathbf{X})d\theta \tag{2.21}$$

Thus, choosing a point estimate for the coefficients in a bayesian regression model depends completely on the loss function $L(\theta, \delta_x)$ defined. When using bayesian inference for regression, where the loss function is the RSS as in ordinary regression (see Section 2.1.5.1), the point estimate to choose is the posterior mean, yielding the prediction points. Therefore, the predictions from a Bayesian Regression model will be

$$\hat{y}_i = \mathbb{E}[y_i|\mathbf{X}] = \int_{y_i} y_i p(y_i|\mathbf{y}) dy_i = \int_{y_i} \int_{\theta} p(y_i|\theta) p(\theta|\mathbf{y}) d\theta dy_i \tag{2.22}$$

where $\mathbf{y}$ is the known, previous data.

2.4.2 Prior distributions

Within MMM, subjective beliefs can prove important, as markets can fluctuate unpredictably. Information about market conditions and sudden events affecting, for example, customers' buying power and in turn sales might not be reflected in the data. Bayesian statistics provides a clear methodology as to how one can incorporate some of these beliefs into the model by using prior distributions. These distributions are arbitrary and chosen by the modeller, and are intended to steer the result towards what is believed to be most credible prior to incorporating the data. However, to achieve advantageous properties of the posterior distributions, such as conjugacy (see Section 2.4.4), one might be forced to choose specific ones. Furthermore, although prior distributions have the intention to incorporate a prior belief, these can be difficult to accurately construct and define. Thus, using a prior that play a minor role in the posterior distribution of the parameters can be convenient [38]. These priors are called *non-informative*, and models constructed with non-informative priors can be used when there is no prior belief. Such a prior is also as a reference point to compare with other models constructed with other priors.

2.4.3 Regularization

Priors can also be used to induce regularization, which can be defined as the choice of including a term to give more stable estimates and avoid overfitting. It is affected by several factors in bayesian modelling [38]. First of all, the location and scale of the prior distribution affects the estimates; a more concentrated prior will have a stronger control of the posterior distribution and thus not only steer more towards a certain value, but also regularize to avoid overfitting to other values. Secondly, the analytical form of the distribution, i.e., which distribution the prior has also affects the regularization as they have different ways of forming the density based on the distance from the most likely value [38].

2.4.4 Markov Chain Monte Carlo

If the posterior distribution is of the same form as its prior, the prior and the posterior are so called conjugate. While conjugate posteriors usually are easy to handle, as they have a clear analytical form, the resulting posterior distribution can often not be tractable to compute in a normal fashion if conjugacy is not the case [38]. To be able to compute these distributions, methods such as Markov Chain Monte Carlo simulations can help forming the posterior distributions. A Markov Chain Monte Carlo simulation is, in short, a simulation in which draws of parameters are performed dependent on one another, forming a joint posterior distribution over the parameters.

A common type of MCMC method the Gibbs sampling (see algorithm 2), which constitutes of sampling from the conditional distributions of each parameter. This results in a simulated distribution, which can then be used to perform Bayesian inference and determine the desired probabilities.

MCMC sampling is a very general technique, used in many bayesian contexts, and help the modeller arrive at posterior distributions which otherwise would not have been possible.

Algorithm 2 Gibbs sampling, as described by Gelman et al. [38]

procedure GIBBS SAMPLING
  Given parameters $\theta_1, .., \theta_K$, initialize $\theta_1^{(0)}, .., \theta_K^{(0)}$
  for t in 1..T **do**
    for k in 1..K **do**
      Sample θ_k from $\theta_k \sim p(\theta_k|\theta_{1:k-1}^{(t)}, \theta_{k+1:K}^{(t-1)})$
    end for
  end for
end procedure

In practice, these will provide similar metrics as to what bootstrap can provide for classical regression models [38]. It is however important to mention that distributions and therefore structures of the data are assumed, which makes it more similar to parametric bootstrapping.

2.4.4.1 Warm-up period (burn-in)

When using MCMC, a common method to use to ensure that the inference is accurate, one commonly uses a method called *burn-in*, also known as the *warm-up period* [38]. This is done to diminish the influence of the starting values, and ensure that one only performs inference in an area of the Markov Chain with the higher probabilities. How many to discard and how to discard depends on the problem, but Gelman [38] is seen to discard about half of all iterations.

2.4.5 Bayesian regression models

The simple linear regression seen in Equation 2.18 provides a simple way to perform Bayesian regression analysis, but has fallbacks due to its simplicity. Two other types of Bayesian regression models with more sophisticated properties are Bayesian hierarchical regression models and Bayesian Structural time-series models, which follow below.

2.4.6 Bayesian Structural time-series Model

While regular Bayesian regression models are very convenient for regression problems where samples are independent, they do not necessarily include and account for time-dependencies. To do this, there are several ways, with Bayesian Structural time-series (BSTS) models as developed by Scott et al. [51] being one of them. A specific package in the programming language R has been developed for this specific type of model, `bsts`[1].

A Bayesian structural time-series model can be considered a so called State Space model, in which the previous state affects the next. It has a hidden (Equation 2.24) and an observed (Equation 2.23) component with latent state variables α_t and ϵ_t and ν_t being noise components.

$$y_t = Z_t^T \alpha_t + \epsilon_t, \quad \epsilon_t \sim \mathcal{N}(0, H_t) \tag{2.23}$$

$$\alpha_{t+1} = T_t \alpha_t + R_t \nu_t, \quad \nu_t \sim \mathcal{N}(0, Q_t) \tag{2.24}$$

Here, R_t, Z_t and T_t are matrices containing known values and unknown parameters giving the relations between the different states. H_t and Q_t are noise component matrices.

To calculate the probability of the coming latent space α_{t+1} and thus perform inference, filtering and smoothing computations are done. While filtering (Equation 2.25) is computing the probability of a state given all the previous states, smoothing is computing the probability of a state given the previous and a number of coming given states (Equation 2.26).

$$p(\alpha_t|y_{1:t}) \propto p(\alpha_t|y_{1:t-1}) p(y_t|\alpha_t) \tag{2.25}$$

[1] Available at `https://cran.r-project.org/web/packages/bsts/bsts.pdf`. Accessed 18/6 - 2019.

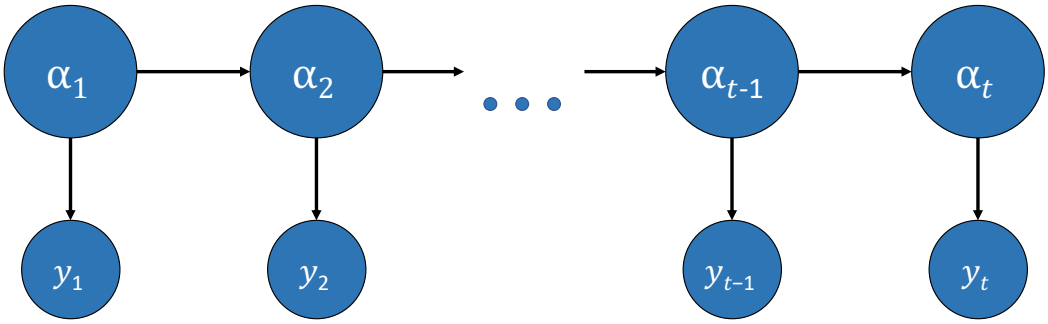


Figure 2.2: A simple example of a state space model. The latent state in the previous timestep affect the next latent state, which affects the observed state.

$$p(\alpha_t|y_{1:T}) \propto p(\alpha_t|y_{1:t})p(y_{t+1:T}|x_t), \quad 0 \leq t < T \tag{2.26}$$

A state space model is a flexible model, and can be extended to contain regressor components, creating the basic structure as seen in 2.27. Therefore, it is able to consider auto-correlated errors, seasonality, trend components and regression components simultaneously [51].

$$\begin{aligned} y_t &= \mu_t + \gamma_t + \beta^T \mathbf{x}_t + \varepsilon_t \\ \mu_t &= \mu_{t-1} + \delta_{t-1} + u_t \\ \delta_t &= \delta_{t-1} + v_t \\ \gamma_t &= -\sum_{s=1} \gamma_{t-s} + w_t \end{aligned} \tag{2.27}$$

Here, μ_t is a trend, γ_t, which has a slope δ_t. τ_t, the seasonal component during each time, is a set of dummy predictors; as many as the number of seasons, decided by the modeller. Although this creates the disadvantage of risking having too many predictors at once, the model still provides the advantage of incorporating all components (seasonal, trend and auto-correlated errors) at once, which otherwise is difficult.

2.4.6.1 Priors and posteriors in a Bayesian Structural time-series model

As seen in the model 2.27, the regression component can be added to incorporate other effects into the state space model. However, as there may be many regressor components, possibly more than the amount recommended. Not more than 1 predictor for every 10 data points is recommended [43], yielding a maximum of about 16 predictors for a three-year series of weekly sales ($156 \approx 160$). To be able to include several at once, possibly more than 16, spike and slab priors on the regression coefficients are used. The Spike and Slab priors allows for either including or excluding parameters into the model by setting them to 0. This allows for

a more sparse model, in which only a few β-coefficients are picked each MCMC-sample, which together forms a posterior distribution for each predictor. The joint posterior distributions are

$$\begin{aligned} p(\beta, \gamma, \sigma_\epsilon^2) &= p(\beta_\gamma|\gamma, \sigma_\epsilon^2)p(\sigma_\epsilon^2|\gamma)p(\gamma) \\ \beta_\gamma|\sigma_\epsilon, \gamma, y^* &\sim \mathcal{N}(\hat{\beta}_\gamma, \sigma_\epsilon(V_\gamma^{-1})^{-1}) \\ \frac{1}{\sigma_\epsilon^2}|\gamma, \mathbf{y}* &\sim Ga\Big(\frac{N}{2}, \frac{SS_\gamma}{2}\Big) \end{aligned} \tag{2.28}$$

where

$$\begin{aligned} N &= \nu + n \\ SS_\gamma &= ss + y*^T y^* + b_\gamma^T \Omega_\gamma^{-1} b_\gamma - \hat{\beta}_\gamma^T V_\gamma^{-1} \hat{\beta}_\gamma \\ V_\gamma^{-1} &= (\mathbf{X}^T\mathbf{X})_\gamma + \Omega_\gamma^{-1} \\ \hat{\beta}_\gamma &= (V_\gamma^{-1})^{-1}(\mathbf{X}_\gamma^T y^* + \Omega_\gamma^{-1} b_\gamma) \\ y_t^* &= y_t - Z_t^* \end{aligned} \tag{2.29}$$

with the priors

$$\begin{aligned} \gamma_k &\sim Bernoulli(\pi_k) \\ \beta_\gamma|\sigma_\epsilon^2, \gamma &\sim \mathcal{N}\Big(b_\gamma, \sigma_\epsilon^2(\Omega_\gamma^{-1})^{-1}\Big) \\ \frac{1}{\sigma_\epsilon^2}|\gamma &\sim Ga\Big(\frac{\nu}{2}, \frac{ss}{2}\Big) \end{aligned} \tag{2.30}$$

The priors on β_γ and σ_ϵ^2 can thus be adjusted through the parameters b_γ, Ω_γ^{-1}, ν and ss. Here, b_γ is the expected value of β, Ω_γ^{-1} sets the strength of b_γ, ν sets the expected value of $\frac{1}{\sigma_\epsilon^2}$ and ss the strength of ν [51].

γ_k are Bernoulli distributed variables used as Spike-and-slab priors, for which it holds that

$$\gamma_k = \begin{cases} 1 & \text{if } \beta_k \neq 0 \\ 0 & \text{if } \beta_k = 0 \end{cases} \qquad \forall k \in P$$

Each draw of γ_k therefore indicates whether to include variable k or not in each MCMC-draw. The prior inclusion probabilities π_k can also be considered **hyperpriors**, prior distributions for a parameter in another prior distribution, that are to be set. The distribution of the priors results in a conjugate posterior distribution, which enables faster and more efficient sampling (see Section 2.4.4). They help form the conjugate posteriors, which can be written as defined by Scott et al. [51] (see Equation 2.28). From these conjugate posteriors, MCMC samples are drawn. In each MCMC draw, a model with a couple of randomly picked regressors are formed, and from this the model is estimated. Thus, through the use of Spike-and-slab priors, a sparse model only including a few variables is created. From all the MCMC draws where a regressor is picked, a posterior distribution of the regressor is constructed.

To forecast with a Bayesian Structural time-series model, one simply forms a predictive distribution as shown in Section 2.4.1. This can then be summarized using a point estimate such as the mean as suggested by Scott et al. [51].

2.4.7 Bayesian hierarchical models

If the parameters in a bayesian model are independent, their prior distribution can be considered to be independent as well. In many cases however, parameters can depend on each other on multiple levels, something not really possible to model in classical regression models. A type of Bayesian regression model, hierarchical models, does however allow these structures by letting the model involve multiple parameters where the parameters are dependent on multiple levels.

When there exists hierarchical data, non-hierarchical models are less appropriate, as they can tend not to fit the data well; either through overfitting if too many parameters are used, or by not distinguishing the differences for the different regions [38]. Hierarchical regression models are therefore useful, provided that there are predictors at different levels in some way [38]. In MMM, hierarchical models can be used to model on different geographical levels. Provided that there exists data on a regional, city or store basis, this can provide a foundation for differentiating between the different regions [66]. Not only does this allow the modeller to obtain inference on differences between regions; it also contributes to partly solving the issue in MMM of having few data points. Creating more complex models might force the modeller to generate more parameters than data points, which generally leads to a highly insecure model. Thus, when having data at a finer level of granularity, it might be convenient to use this data, such as on a regional or city basis [66].

Hierarchical models can be visualised through directed acyclic graphs (DAGs), in which the parameters are the edges, and their influence on other parameters the vertices. An example of a hierarchical linear model as proposed by P. Rossi et al. is given below in equations 2.31 and 2.32. The DAG is given in Figure 2.3. This model allows to divide the data according to some rule, e.g., a geographical, into k different parts, where $y_{i,t} \in \{1, .., k\}, \{1, .., T\}$ can be used to predict the response for each different division. The model is formulated as following on a geographical basis:

$$\begin{aligned} y_{i,t} &= \mathbf{X}_{i,t}\beta_i + \varepsilon_i \\ \varepsilon_i &\sim \mathcal{N}(0, \tau_i) \\ \beta_i &\sim \mathcal{N}(\Delta, V_\beta) \end{aligned} \tag{2.31}$$

where $y_{i,t}$ is the response of geographic location i. By then choosing the conjugate hyper-priors, these can be stated as

$$\begin{aligned} \tau_i &\sim \nu_i \text{ssq}_i / \chi^2_{\nu_i} \\ V_\beta &\sim IW(\nu, V) \\ \Delta &\sim \mathcal{N}(\bar{\Delta}, V_\beta \cdot A^{-1}) \end{aligned} \tag{2.32}$$

where

- τ_i the variance of the each geographic location i
- Δ is the mean of the geographic coefficients.
- V_β is the covariance of the geographic coefficients.
- ν_i is the certainty in the prior variance of each geographic location i.
- ssq_i is a scaler for the variance for each geographic location i.
- V signifies the mean of the covariance matrix V_β in conjunction with ν which also signifies the certainty in the mean.
- $\bar{\Delta}$ is the prior mean of Δ.
- A signifies the certainty in the prior mean $\bar{\Delta}$.

This model can be used to achieve a finer granularity of the prediction, and create a join model, for example, on a store- or regional basis within MMM.

2.4.8 Previous Bayesian MMM modelling

Previously, several papers have tried a Bayesian approach to MMM modelling. Liu et al. [53] performs a MMM study using MCMC methods in a simplified setting only with simulated data, where the sales only depend on time and media channel spendings. Their simulations showed that the algorithm created could provide a base for calculating a new allocation that

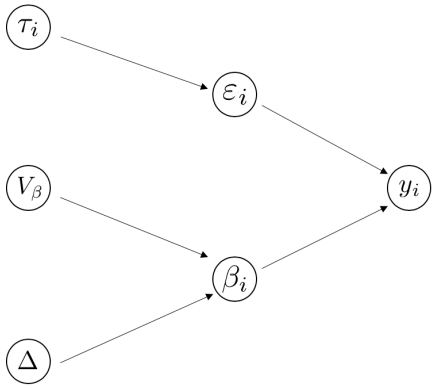


Figure 2.3: The linear hierarchical model as proposed by P. Rossi et al.

Source: P. Rossi et al. [66]

could increase revenues by 60 per cent on the simulated data. They do however critique their own simplifications, as they admit that there is a possible non-linear relationship between advertising effect, as well as there are other factors not accounted for in the model that clearly affect the sales outcome, such as the competition's advertising [53].

Further, in 2017, Jin et al. [48] attempts to apply a Bayesian regression model to data retrieved from a shampoo advertiser. Here, data such as spend on major channels and other predictors such as price per ounce, All Commodity Volume (ACV) weighted product distribution and promotions are included in the model, although not macro-economical factors such as weather, Gross Domestic Product (GDP) and consumer confidence index (CCI). The response variable was modelled as

$$log(y_t) = \tau + \sum_{m=1}^{M} \beta_p \text{Hill}(x^*_{t,m}; \mathcal{K}_m, \mathcal{S}_m) + \sum_{c=1}^{C} \gamma_c z_{t,c} + \epsilon_t, \quad \epsilon_t \sim \mathcal{N}(0, \sigma^2) \tag{2.33}$$

where $x^*_{t,m}$ was the media spend predictor, processed through a decay rate function, τ the baseline sales, γ_c the effect of control predictor z_c and Hill a function modelling the relation between media-predictors and the explanatory variable. To achieve the parameter estimates, the posterior distribution was given as

$$p(\Phi|\mathbf{y}, \mathbf{X}) \propto \mathcal{L}(\mathbf{y}|\mathbf{X}, \mathbf{Z}\Phi)p(\Phi) \tag{2.34}$$

where $\mathcal{L}(\cdot)$ is the likelihood function. The parameter estimates obtained had a larger variance, resulting in a large variance when deciding the optimal combination of media spending. Half-normal, normal and uniform priors were used on the β parameters and the $\mathcal{K}$, for which a beta-distribution or uniform distributions were used. The results showed that the different priors achieved results highly differing, thus showing the importance of a good informative prior. The authors thus stressed the importance of good data and gathering of expert information.

As previously discussed in Section 2.4.7, hierarchical models can be utilized to handle data when there exists information at a finer, geographical granularity. Sun et al. does exactly this, by dividing the data on a sub-national level when possible [69]. They apply a Bayesian model, identical to the Bayesian model created by Jin Y. et al on each geographical data set, creating a Geo-level Bayesian Hierarchical modelling (GBHMM), with same priors as in a previous work from Google [48] to make them comparable. This model resulted in tighter credible intervals on the parameter estimates, compared to the regular model having data only

on a country basis. This due to the effectively larger sample size to sample from, reducing the importance of the prior, therefore reducing the bias as well [69]. One might say that the data was augmented, as the increase in data points increases the amount of data, which could be an explanation of the reduced uncertainty.

2.4.9 Critique towards bayesian inference

As mentioned previously in Section 2.4, bayesian inference allows the modeller to incorporate subjective beliefs for the parameters in the model. This can many times, as D. McNeish explains, lead to better results when having a small dataset [58]. Thus, it can help mitigating the issue of small datasets, a widely known issue within MMM. One must however address the issue and importance of using Bayesian Modelling responsibly. Incorporating a subjective belief means that one introduces an intentional bias to the model, which can yield a misleading model if used incorrectly. He further discusses several conditions that must be met if one wishes to utilise Bayesian methods. The importance of using a prior derived from reason is stressed, for example, through prior or expert information, and not using, for example, a standard prior issued by the software used for simulation, demonstrating the importance of this specifically for multi-level models such as hierarchical models with a small amount of samples in each cluster (see Section 2.4.7).

2.5 XGBoost

As mentioned previously, one usually uses parametric regression models in an MMM context due to the possibility of specifically attributing impact on the response variable to specific predictors. However, with new possibilities of interpreting non-parametric regression models, possibilities to use these in a MMM setting has arisen.

XGBoost is a non-parametric method recently gaining popularity in the Data Science community for solving many different types of machine learning problems in different settings. It is a non-parametric tree boosting method combining multiple decision trees [20] and can be used for both classification and regression problems. It has been the best performing method for several different regression tasks, among those include store sales prediction [20].

In order to understand XGBoost, one must first understand regression tree models, also known as CART [6]. CART models take the shape of decision trees based on the values of the predictors. In each node, there is a decision criteria, determining which path down the decision tree the data point will take. Each leaf has a value or class, and once a data point reaches a leaf, the point is assigned the value or class corresponding to the leaf. In this way, CART models partition the input predictor hyper-dimensional cuboids along the predictor axes. In regression trees, the optimal predictive value for each region $\mathcal{R}_\tau$ is given by Equation 2.35. In other words, it will be the mean of all the training points' response values inside the cuboid.

$$y_\tau = \frac{1}{N_\tau} \sum_{t_i \in \mathcal{R}_\tau} t_i \tag{2.35}$$

where τ denotes which region $\mathcal{R}_\tau$ within the solution space, N_τ denotes the number of points within the region and t_i is the label of a training point located within the region $\mathcal{R}_\tau$. Do note that this might not be the case in the case of multiclass-classification regression trees.

Now, consider a regression problem where $y_{1:n}$ is the response variable data, $\mathbf{X}_{1:n}$ being the predictors. XGBoost, creates a group $\mathcal{F}$ of trees

$$\mathcal{F} = \{f_k(\mathbf{x}) = w_{q_k(\mathbf{x})}\}, \quad (q_k : \mathbb{R}^m \to T, w \in \mathbb{R}^T, k \in \{1, ..., N\})$$

where q_i is the structure of tree i, and N being the number of trees. The sum of weights

$$\hat{y}_i = \phi(\mathbf{x}_i) = \sum_{k=1}^{K} f_k(\mathbf{x}_i) = \sum_{k=1}^{K} w_{q_k(\mathbf{x})}, \quad f_k \in \mathcal{F} \tag{2.36}$$

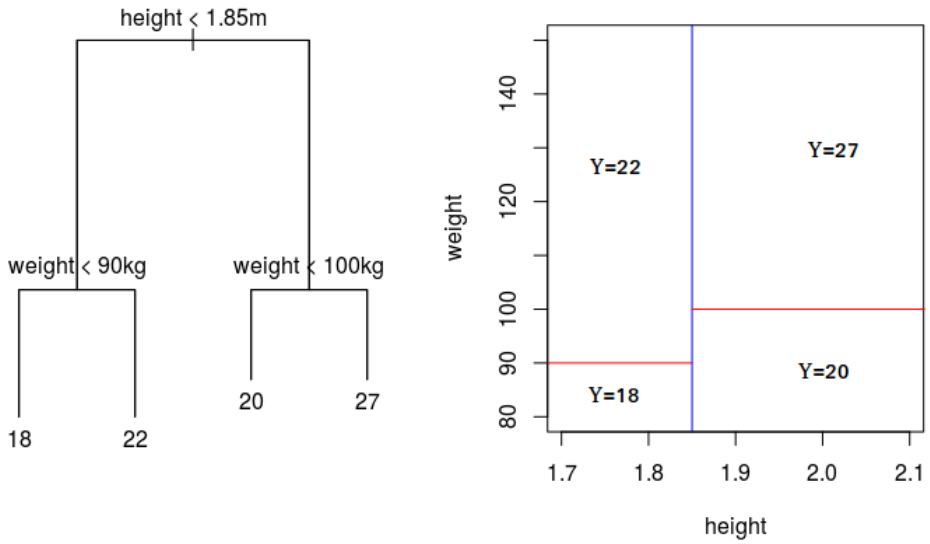


Figure 2.4: Example of a regression tree deciding how many points a basketball players will score in a game based on their weight and height.

Source: insightr [47]

obtained from the trees accounts for the prediction result, as seen in Equation 2.36. The objective function $\mathcal{L}(\phi)$ to optimize the tree structures used is then

$$\mathcal{L}(\phi) = \sum_{i=1}^{n} l(\hat{y}_i, y_i) + \sum_{k=1}^{K} \left(\gamma N + \frac{1}{2}\lambda \|w_k\|^2\right) \tag{2.37}$$

where $\hat{y}_i$ is the predicted value for the data point in question and y_i is the actual value. γ and λ can be considered regularizing hyper parameters. This objective function cannot be optimized with traditional methods, which means that the model has to be optimized in additive manner.

A simple and hypothetical example of an XGBoost algorithm can be found in Figure 2.5. Here, the response variable represents the amount of hours a person spends at a computer per day. The family of trees $\mathcal{F}$ can be considered to be

$$\mathcal{F} = \{f_{tree1}(\mathbf{x}), f_{tree2}(\mathbf{x})\} \tag{2.38}$$

where

$$\begin{aligned} f_{tree1}(\mathbf{x}) &\in \{4,\ -1,\ 6\} \\ f_{tree2}(\mathbf{x}) &\in \{-1,\ 1\} \end{aligned} \tag{2.39}$$

where the outputted weight from each tree will depend on the input features. Assume we have a 35 year old office worker without a university education. Given the hypothetical model, the prediction will be

$$\begin{aligned} \hat{y}_i = \phi(\mathbf{x}) &= \phi(Age = 35, \text{Office worker} = Yes, \text{University education} = No) \\ &= \sum_{k=1}^{K} w_{q_k(\mathbf{x})} = 6 + (-1) = 5 \end{aligned} \tag{2.40}$$

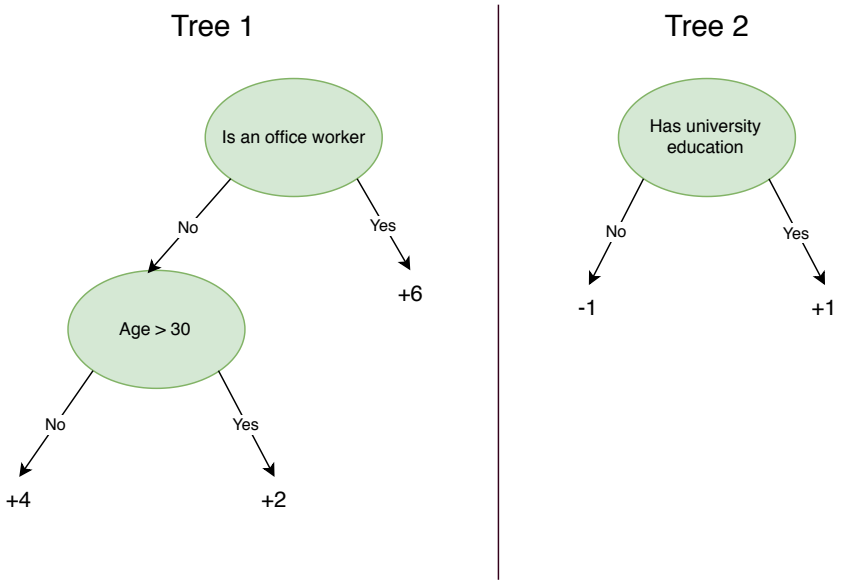


Figure 2.5: Hypothetical example of XGBoost. The response variable corresponds to hours spent at the computer on a daily basis.

leading to a prediction of 5 hours spent in front of the computer per day.

To avoid overfitting in XGBoost, two methods are used. [20] First of all, shrinkage through regularisation parameters reduces the influence of each individual tree to leave room for coming trees to have influence on the outcome. Secondly, column sub-sampling is used, which means that one chooses a subsample of columns for each tree. This reduces the similarity of the different trees, thus preventing overfitting.

2.6 Explaining models using additive feature attribution methods

Although regression trees individually are highly interpretable, XGBoost can consist of an arbitrary number of CART-trees, making the interpretation of the model as a whole difficult. There are however several ways to interpret the complete model; to interpret XGBoost, one can use methods such as **Additive Feature Attribution Method** (AFA) (see Definition 5). These can be defined as *"any interpretable approximation of the original model"* [56, p. 2].

Definition 5. *Additive feature attribution methods*
Let $z' \in \{0,1\}^M$ *be a set of binary features, stating if feature* x' *is observed or not. If the method is an Additive feature attribution method, the explanation model can be considered to be*

$$g(z') = \phi_0 + \sum_{i=1} \phi_i z'_i$$

where M *is the number of simplified input predictors and* $\phi_i \in \mathbb{R}$ *[56].*

A big problem when interpreting complex ensemble methods such as XGBoost is that AFAs are inconsistent, meaning that when a model is changed to make a predictor have a higher impact on the outcome, the methods will still lower the importance of the predictor in question. One recent AFA method that can help determine which predictors are the most significant is the Shapely Additive Explanations (SHAP), claimed to actually be consistent [56]. Theoretical results shows that there is a unique solution for every problem with 3 desirable properties for SHAP, which previously was proven for classical Shapley value estimation methods [56]. The first property is **Local Accuracy**, meaning that simplified input x' of x will have an output

$f(x')$ matching $f(x)$. The second desired property, known as **Missingness** is that predictors missing have no impact. The third desired property, and possibly the most important is (as mentioned previously) **Consistency** (see Definition 6) [56].

Definition 6. *Consistency*
Let $f_x(z') = f(h_x(z'))$ *and* $z' \setminus i$ *denote setting* $z'_i = 0$. *For any two models f and f', if*

$$f'_x(z') - f'_x(z' \setminus i) \geq f_x(z') - f_x(z' \setminus i)$$

for all inputs $z' \in \{0,1\}^M$, *then* $\phi_i(f', x) \geq \phi_i(f, x)$ *[56].*

SHAP has its origin in the Shapley Value method from game theory, and assigns each predictor a value of importance for each prediction by trying to obtain the Shapley values of the conditional expectation function of the original model.

Definition 7. *Linear SHAP*
Let $y_t = \mathbf{X}\beta = \beta_0 + \sum_{i=1}^{p} \beta_i x_{i,t}$. *The SHAP values can then be approximated as*

$$\phi_0 = \beta_0, \quad \phi_i(y_t, x_{i,t}) = \beta_i(x_{i,t} - \mathbb{E}[x_{i,t}])$$

where the predictors $x_{1,t}, .., x_{p,t} \quad \forall t \in T$ *are independent [56].*

SHAP values can be used for many different models [56]. For linear models, Linear SHAP (see Definition 7) can be used, where one assumes predictor independence of the predictors, the parameters of the model are used directly to calculate the SHAP values, providing a relation between models and an intuition of what the SHAP values actually represent. SHAP values show the contribution of each predictor for each prediction, while Shapley values are weights used to weigh the relative importance of each feature. However, to compute the exact SHAP values can be challenging computationally-wise in many situations, and approximations often has to be used for most models. However, for XGBoost and other tree ensemble models, a specific method called TreeSHAP can be used to compute the exact values in quadratic time [55] with respect to the maximum depth on any tree. SHAP values are defined as in Definition 8.

Definition 8. *SHAP-values*
Let S *be the set of non-zero indexes in* $z' = \{z_1, .., z_m\}, z_i \in \{0,1\}$ *and* $\mathbb{E}[f(x)|x_S]$ *be the expected value of the function conditioned on S. The conditional expectations can be then be defined as*

$$f_x(S) = f(h_x(z')) = \mathbb{E}[f(x)]$$

The SHAP values $\phi_1, .., \phi_M$ *is then defined as*

$$\phi_i = \sum_{S \subseteq N \setminus \{i\}} \frac{|S|!(M - |S| - 1)!}{M!} [f_x(S \cup \{u\}) - f_x(S)]$$

where N *is the set of all input predictors and* M *is the number of predictors. [55]*

2.7 Uncertainty within regression

There are multiple ways in which to view uncertainty. Here it will be defined as uncertainty in the correctness of a model. As Pox et al. [9] stated in 1986, all models are wrong due to their simplification of reality, but can still be useful as they can yield insightful results. In a MMM context, the regression function $F(\cdot)$ is chosen by the modeller, and modelling this function that predicts sales for a certain geographical region and period of time is a complex task due to many reasons, as many authors have acknowledged before [63, 19, 25]. Not only is the data limited or often not granular enough [63], there might often be several models that

fit the data well as exemplified by Chan [19]. The choice of model within demand studies does not have a deterministic answer; in fact, Chan claims that several different models with completely different underlying architectures might fit the data equally well, yielding models giving completely different ROI values, the most important outcome of MMM. However, when investigating some models further by looking at the reasonableness of the coefficient values, or looking at the certainty of the parameter estimates, some models can be ruled out.

In order to pick the correct model out of the highly explaining models, there exists several methods. An important part of MMM and the selection of the model is to thoroughly investigate the model's results on its interpretable parts and, if possible, put restrictions and constraints preventing well fitted but unreasonable models to occur.

2.7.1 Multicollinearity

One subdomain of uncertainty is that which rises from multicollinearity. Multicollinearity occurs when several predictors are highly correlated, and it becomes hard to determine which effect come from which predictor. If the only task of the model is to provide a good fit, multicollinearity is not an issue, as a highly multi-collinear model can still provide small errors [64, 19]. However, if interpretation is of importance, as in MMM, multicollinearity poses to be an issue of uncertainty, as it makes it more difficult to determine which predictor contributes how.

$$R^2 = \frac{\sum_{i=}^{n}(\hat{y}_i - y_i)^2}{\sum_{i=1}^{n}(y_i - \overline{y})^2} \tag{2.41}$$

One way of detecting multicollinearity is to look at the conditional number of the data matrix $\mathbf{X}$, stated as

$$\kappa(\mathbf{X}) = ||\mathbf{X}||_2 \cdot ||\mathbf{X}^{-1}||_2 \tag{2.42}$$

Williams [75] states that an informal rule of thumb is that a conditional value of the matrix $\mathbf{X}$ higher than 15 suggests that multicollinearity is present, and a serious concern if it is greater than 30. Another way of measuring multicollinearity is to use the Variance Inflation Factor (VIF) [64]. The VIF-value for each predictor is given by Equation 2.43, where R_j^2 is the R^2-value of the regression for which the other predictors are fitted with an OLS-regression to the predictor x_j. The R^2 falls between 0 and 1 and can be seen as the percentage of the variance of the explanatory variable which is explained by the predictors. In this case, the percentage of variance of x_j explained by the other variables. A VIF-value larger than 10 suggests that multicollinearity is a problem for the predictor [64]. This occurs if the data matrix $\mathbf{X}$ is singular or nearly singular, and gives a view on which variables are affected the most by this.

$$VIF_j = \frac{1}{1 - R_j^2} \tag{2.43}$$

To mitigate multicollinearity, several methods exist. One way is using priors in Bayesian regression models [38], regularizing the coefficients. Another is Shapley Value Regression (SVR), presented below.

2.7.1.1 Shapley Value Regression

Shapley Value Regression is a method to determine the relative importance of predictors and adjusting the respective coefficients accordingly. Given a set of predictors $\{X_1, X_2, ..., X_p\}$, let $S_{j,-i}$ be the set of all combinations of predictors of size j excluding i. Using the ordinary least squares estimator, the relative importance can be measured as seen in equations 2.44 and 2.45.

$$SV(X_i) = \frac{1}{m}\sum_{j=0}^{m-1} SV_j(X_i) \tag{2.44}$$

$$SV_j(X_i) = \frac{1}{|S_{j,-i}|} \sum_{s' \in S_{j,-i}} R^2_{s' \cup X_i} - R^2_{s'} \tag{2.45}$$

The value $SV_j(X_i)$ can be interpreted as the expected contribution of predictor X_i given a uniform distribution over $S_{j,-i}$. The Shapley value, SV_j, can then be interpreted as the expected contribution given a uniform distribution over the amount of predictors. Given this, the coefficients $\beta^{shapley}$ can be obtained through

$$\beta_i^{shapley} = \frac{SV(X_i)}{(X^T X \beta^{ols})_i}, \quad \forall i \in \{1, .., p\} \tag{2.46}$$

where, β^{ols} are the coefficient estimates according to the ordinary least squares. However, Lipovetski and Conklin [52] instead suggest the formulation

$$\beta_i^{shapley}(2X^T X \beta^{ols} - X^T X \beta^{shapley})_i = SV(X_i) \tag{2.47}$$

which can be minimized with a set of quadratic equations. With the expected, relative contributions, one can determine which predictors are highly contributing and which are not. After this, coefficients to predictors not relevant to the model will be reduced in sizes, relative to their importance.

2.7.2 Outliers and heavy-tailed distributions

An issue within MMM and many areas of modelling is how to account for outliers in the data. Sales data can, as discussed in Section 2.8.1, vary greatly due to the factors not known or available to the modeller. The data noise might follow a so called heavy-tailed distribution, having a larger chance to draw large values, or simply contain outliers distorting the fit of the model. Both violates a regression model's assumption of normally distributed errors, which creates a higher uncertainty when using OLS.

The OLS loss function, the most basic of linear regression loss functions, has the objective to minimize the function given in Equation 2.48, where $(y_i - \mathbf{x}_i^T \beta)^2$ is the residual for data point i. This makes the effect of a residual grow at a quadratic rate, making the residuals' effect grow quadratically from the distance from the estimating line. If this is not accounted for, the fit of the model might be misleading due to the model overfitting to the outliers, and lead to a higher uncertainty in the model.

$$min \sum_{i=1}^{n} (y_i - \mathbf{x}_i^T \beta)^2 \tag{2.48}$$

There are several methods to define which points are considered outliers, but one common definition is to use Tukey's rule, stating that all points outside the 1.5 interquartile range (1.5IQR) can be considered outliers [71].

2.7.2.1 Robust regression

To mitigate the uncertainty that outliers and heavy-tailed distributions can cause, Peter J. Huber [45] proposes to, instead of using the OLS loss function (Equation 2.48), obtain the parameters by minimizing the sum of the residuals (see Equation 2.49) sent in to the Huber loss function, given in Equation 2.50.

$$\hat{\beta} = argmin_{\hat{\beta}} L(\hat{\beta}) = argmin_{\hat{\beta}} \sum_{i=1}^{n} \rho_c\Big(\frac{y_i - \mathbf{x}_i^T \hat{\beta}}{\hat{\sigma}}\Big) \tag{2.49}$$

Using this for the loss function yields an estimate known as the M-estimate. Here, $\hat{\sigma}$ is an error scale estimate to scale down and normalize the residuals. The result of the Huber loss functions is that the loss function behaves in the same way as the OLS loss function for points

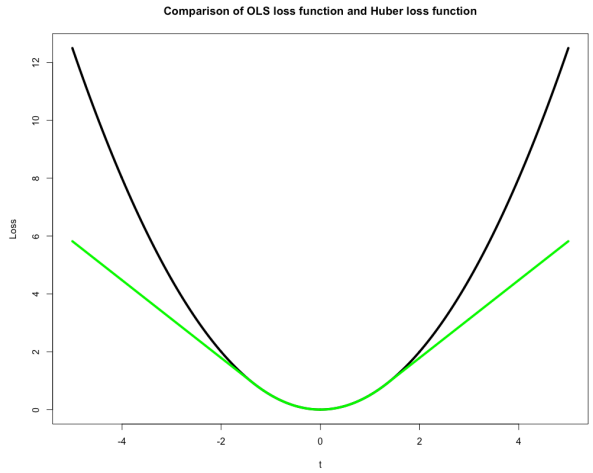


Figure 2.6: Plot showing the difference between OLS losses and Huber losses. After $|t| \geq c$, the loss function starts growing linearly.

lying near their predicted values, but start growing linearly instead of quadratically after a certain point c. This reduces the effect of points lying far away from their predicted values, making it significantly less sensitive to outliers [78]. For a visualisation of the loss functions, see Figure 2.6.

$$\rho_c(t) = \begin{cases} \frac{1}{2}t^2 & \text{for } |t| < c \\ c \cdot |t| - \frac{1}{2}c^2 & \text{for } |t| \geq c \end{cases} \tag{2.50}$$

Many other robust methods exist to reduce the overfitting of OLS estimators. An extension of the M-estimate is the MM-estimate, which according to Yu et al. [78] is significantly better in terms of efficiency and breakdown point. The MM-estimate is a three-step procedure, in which an initial estimate $\hat{\beta}_0$ is first computed with an M-estimate. This is followed by computing a robust M-estimate for the $\hat{\sigma}$ scaling parameter in the second step. After this, in the final step, the parameter estimate $\hat{\beta}_{MM}$ is retrieved by computing an M-estimate of it, starting at $\hat{\beta}_0$. V. Yohai [77] showed theoretically that

$$L(\hat{\beta}_{MM}) \leq L(\hat{\beta}_0)$$

holds, thus proving its superiority to the M-estimate in terms of loss. The study by Yu et al. [78] also showed empirically that the MM-estimate outperformed M-estimates in several settings. In particular, in the case of having high-leverage outliers and outliers in the y-direction, the M-estimates was at times worse by as much as an order of magnitude. However, in most other settings, the difference was not great, although MM-estimates had a lower MSE on the parameter estimates in most cases.

2.8 Marketing Mix Modelling

Due to MMM models' high focus on interpretability, the most common approach is to model the sales through a regression model as referred to in Section 2.3.1. The functional form can be seen in Definition 9. Regression models like these allow the modeller to interpret the contributions of each media spending through the estimates of the parameters corresponding to each marketing channel. A crucial factor to account for is that these estimates are not only precise and fit the data well, but are also confident in explaining the actual contributions. This can often pose to be an issue due to multicollinearity, a common problem described in Section 2.7.

Definition 9. *Marketing Mix Model*
Let $\mathcal{D} = (\mathbf{y}, \mathbf{X})_{1:T}$ *be the data, where* $\mathbf{y} = \{y_1, .., y_T\}$ *denotes the sales and* $\mathbf{X} = \{\mathbf{x}_1, .., \mathbf{x}_T\}$ *denote the features chosen to be incorporated into the model. Further, let* $\mathbf{z}_t = \{\mathbf{z}_{t,1}, .., \mathbf{z}_{t,C}\}$ $\forall t \in T$ *denote the vectors of control variables incorporated into the model,* $\mathbf{\Phi}$ *be the parameters of the model and* $\hat{\mathbf{\Phi}}$ *be the model's hyperparameters. A general Marketing Mix Model can then be defined as*

$$\hat{y}_t = F(\mathbf{x}_{t-L+1}, ..., \mathbf{x}_t, \mathbf{z}_{t-L+1}, .., \mathbf{z}_{t-L+1}; \mathbf{\Phi}, \hat{\mathbf{\Phi}})$$

where L is the longest time that the control variables and previous features are assumed to have a lagged effect, and $F(\cdot)$ *is a function chosen by the modeller [19].*

2.8.1 Issues within MMM

As mentioned in the introduction, there are several challenges within MMM that needs to be addressed when building a MMM model.

First of all, the number of factors affecting sales of a product on national levels is large. This makes it hard to include and account for everything affecting the response variable, yielding a large noise component in whichever model one chooses.

Second of all, MMM data sets generally have few data points. In most cases, one uses data on a country- and weekly- or monthly basis [48, 19]. A three year history of data on a weekly basis would then only yield a maximum of 156 ($3 \cdot 52$ weeks) data points in total. This yields the issue of making it hard to fit a model appropriately, as many models requires large amounts of data to converge to a reliable, robust result. A possibility to extend this is to include sales data from earlier years, but some researchers within econometrics claim this to lead to misleading conclusions. This is widely known as the Lucas critique, which states that it can prove highly misleading to base economical policies on econometrical models only incorporating historical data, in particular with larger volumes of aggregated data [54]. Therefore, including more data might rather lead to more inaccurate estimates and inferences, which limits the amount of data, often causing a high amount of uncertainty in the models.

Third of all, information that cannot be reflected through the data, and the fact that historical data can actually point in the wrong direction due to fluctuations in the market make it crucial to be able to incorporate additional information into the model.

Lastly, as in most economical data shown through a time-series, there might exist time-dependencies between the data points. The sales in a certain week might influence the sales in the next, and seasonal and trend components are also widely recognised to exist [41]. and in order to create a model as perfect as possible and achieve a thorough analysis, it is important to extract these components from the model and account for these, to then obtain the estimates of the media contributions after or while these components are extracted.

The non-linearity of marketing expenditures, known as the *Shape effect*, is also an issue that has to be considered. The Shape effect, defined by Tellis [70] as the *"change in sales in response to increasing intensity of advertising in the same time period"* [70, p. 507], refers to the fact that there is a saturating effect in media spending. After a certain amount of spending on a specific channel, as well as media spending in total, the effect becomes smaller. Tellis also claims that an S-like curve is the most reasonable, as a low amount of marketing is most likely drowned out in the noise, while marketing after a while tends to lose its effect once it has reached all targeted consumers. On the other hand, marketing through many different marketing channels tends to have a synergy effect; the more that is spent in different marketing channel, the higher is the increase of sales [69]. The S-shape of the curve has also been criticised and claimed to have little empirical support. Instead, Hanssens [41] et al. claims that most empirical evidence supports a concave response function.

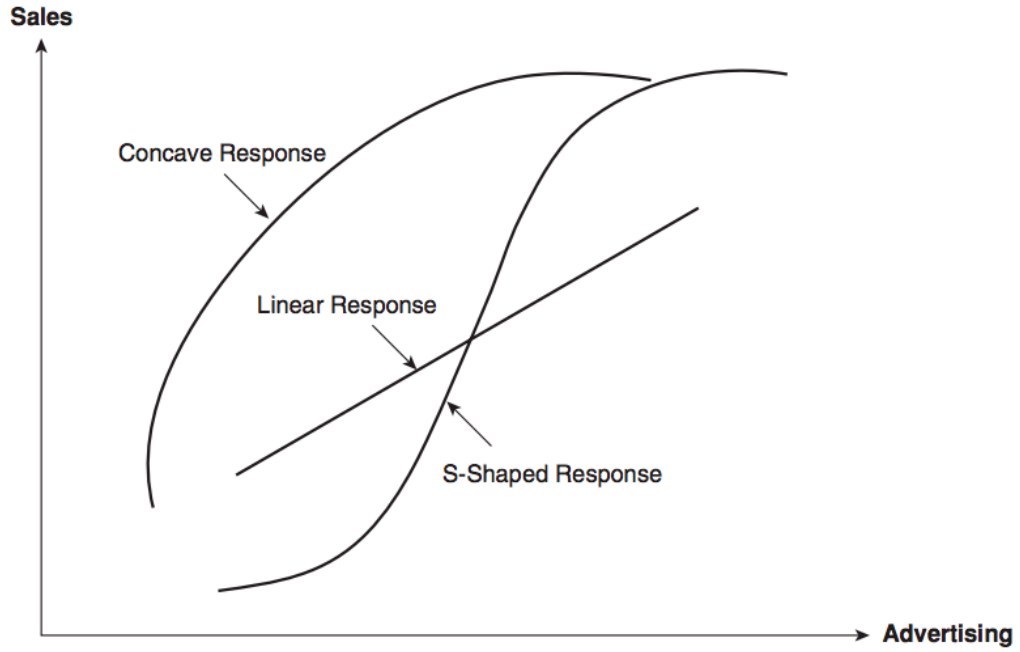


Figure 2.7: Examples of how media spending affects the sales. Tellis claims the S-shape to be the most reasonable response curve (also the source of the figure).

Source: Tellis [70]

$$a_{j,t} = \sum_{j=min\{1,t-L\}}^{t} \lambda^{t-j+1} \cdot x_{i,j} \tag{2.51}$$

2.1: Simple decay rate. L is the maximum amount of time units to spread out the media. λ is chosen appropriately according to specific industry standards for each data set.

2.8.1.1 Decay rates - transforming the media variables

Advertisement spending are often allocated sparsely, as investments are often made in huge chunks. In order to capture some of the properties of the effects of the media spending on sales, one might need to transform the media spending variables to create less sparse data, which rather creates a more realistic distribution of the spending in relation to its effect. One common way to do this is through decay rates [25], in which the spend from one week is distributed among the week it was spent and the coming L weeks. There are several ways to do this, but one common way is through the Simple Decay Rate function, seen in Equation 2.1. An intuition of its effect can be seen in Figure 2.8.

2.8.2 Functional forms of MMM

The MMM response function, previously defined in Definition 9, can be specified in several ways using a parametric regression model, addressing different properties of the data, proving useful in different situations. Hanssen et al. [41] presents a number of different functional forms. First of all, he claims that the linear model, as given in 2.7 is not accurate, but is convenient to apply as it is easy to use and provides a reasonable approximation of the non-linear sales function; in particular when data is only available over a limited period of time.

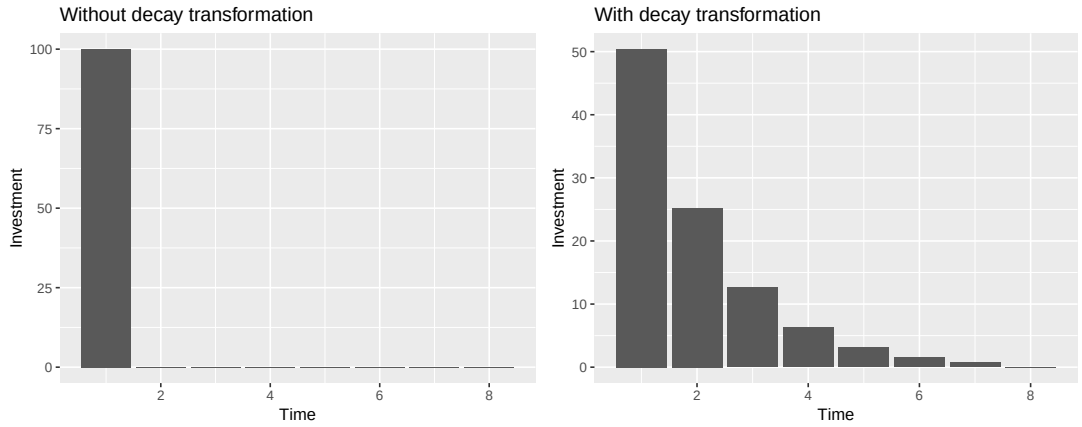


Figure 2.8: Example of spending in a media channel before and after applied decay rate.

It may provide local conclusions, but the model will not be applicable to vastly differing and highly varying data.

A second functional form presented by Hanssens, which addresses the diminishing returns property that market responses are known to exhibit, is the log-linear model. It can be seen as

$$y_t = \beta_0 \cdot \prod_{i=1}^{p} x_{i,t}^{\beta_i} \tag{2.52}$$

which can be expressed in a linear form as

$$\log(y_t) = \log(\beta_0) + \sum_{i=1}^{p} \beta_i \log(x_{i,t}) \tag{2.53}$$

which has the advantage that the coefficients of the predictors can be directly interpreted as the elasticity, thus giving a one-unit increase. This functional form not only provides a concave response as seen in Figure 2.7, but it also allows the diminishing returns to scale, provided that the β-coefficients are between 0 and 1 [41].

A third example of a functional form possible to use is the form given in 2.54 [41] which we shall call the semi-logarithmic model. While this does not address the diminishing returns as it is not concave, this provides the advantage of increasing returns to scale due to the synergy effect spending in different marketing channels tends to have [69].

$$y_t = \beta_0 \cdot e^{\sum_{i=1}^{p} \beta_i x_{i,t}} \cdot \varepsilon \tag{2.54}$$

Both the second and the third functional form have the advantage that the error increases as the predictors increase in value; the bigger the market, the bigger will the fluctuations be.

To summarize, the linear model can provide a reasonable, local estimate, while functional forms addressing properties of sales responses can, during the right circumstances, provide more general models over a wider range of predictor values [41].

2.8.3 Obtaining ROI-estimates in a MMM-model

As mentioned previously, the main outcome of an MMM model is to determine the ROI on the media spend predictors. There are ways of doing this in aggregating the ROI into a constant value. However, a constant ROI would imply a linear model, since it suggests that increasing

the spending by x in one channel would increase the sales by the ROI times that amount. As a result, the ROI of a linear model can be described as seen in Equation 2.55.

$$ROI_m = \frac{\sum_{t=1}^{T} \beta_m \cdot x_{m,t}}{\sum_{t=1}^{T} x_{m,t}} = \frac{\beta_m \cdot (x_{m,1} + ... + x_{m,t})}{(x_{m,1} + ... + x_{m,t})} = \beta_m \tag{2.55}$$

In other words, the ROI of a media channel is the β-coefficient for the media channel spend predictor.

For the log-linear and the semi-logarithmic functional forms, the linearized ROIs are slightly more complicated to calculate. Garg et al. [27] presents a method that determines these ROIs for each media channel based on dividing their total normalised contributions C_m by their total spend, $S_m = \sum_{t=1}^{T} x_{m,t}$. A further motivation to as to why to calculate the contributions like this can be seen in their work. This method thus provides a linearization and thus a simplification of the ROI function, that is in theory considered to rather have a concave response function as claimed by Hanssens [41] or S-shaped as claimed by other authors [19, 48, 69, 70].

Definition 10. *Assume a model where*

$$y_t = \beta_0 \Big(\prod_{i \epsilon J} x_{i,t}^{\beta_i} \Big) \cdot \Big(\prod_{i \epsilon L} exp(x_{i,t}\beta_i) \Big) \cdot \varepsilon$$

for some set of predictors J and L, where $x_{i,t}$ denote the predictor i's value at time t. Further, let $C_{i,t}$ denote the contribution from predictor m during time period t. The normalised contribution $C'_{i,t}$ can then be calculated as

$$C'_{i,t} = C_{i,t} + \frac{C_{i,t}(V_t - V'_t)}{V'_t}$$

where V'_t denotes the total regressor contribution, given as

$$V'_t = \sum_{i=1}^{p} |C_{i,t}|$$

and

$$V_t = \sum_{i=1}^{p} C_{i,t}$$

The ROI for each media channel m can then be calculated as

$$ROI_m = \frac{\sum_{t=1}^{T} C_{m,t}}{S_m}$$

where $S_m = \sum_{t=1}^{T} x_{m,t}$.

While this way of measuring the linearised ROI values, they do not reflect the true ROI function of the non-linear models but merely a summarisation. As a result, the validity of such a method can be criticised. It can also be criticised in the sense that theoretical guarantees have not been presented to guarantee that the way of calculating the contributions is consistent. Jin et al. [48] proposes a different, more general method, in which the ROI is calculated as

$$ROI_m = \frac{\sum_{t=t_0}^{t_1} F(x_{m,t-L+1}, ..., x_{t,m}; \Phi) - F(\tilde{x}_{m,t-L+1}, ..., \tilde{x}_{m,t}; \Phi)}{\sum_{t=t_0}^{t_1} x_{m,t}} \tag{2.56}$$

where $\tilde{x}_{t,m}$ is the spend change, and t_0, t_1 is the change period. This is also a linear summarisation, but has two disadvantages. First of all, it assumes that no media spending affect the outcome of another spending, which is not true in the general case. Secondly, it examines hypothetical outcomes over a period of time by assuming hypothetical spendings $\tilde{x}_{t,m}$, something that the method provided by Garg et al. do not. Neither this have been provided with theoretical guarantees.

2.9 Time-series analysis

A time-series is a series of data recorded over time, usually with an equal amount of time between points. A time-series is an observation of a stochastic process, a statistical concept which evolves with time [10]. The analysis of a time-series is thus the analysis of an observation, trying to capture the structure of the underlying stochastic process. The data in Marketing Mix Modelling is a time-series as it is recorded over time. As a result, the tools developed for time-series analysis can be useful in the modelling of the marketing mix. While time-series can take many structures, only discrete time-series that are equidistant in observations will be considered here.

2.9.1 Stationarity

One fundamental difference compared to uncorrelated data is that a realisation $\{X_t : t \in T\}$ of a stochastic process only is one observation. This makes estimation more difficult and it is important that this process has a clear structure if inference is to be done. Since there will normally only be one realisation, and not multiple i.i.d. copies, the assumption of stationarity becomes crucial [67]. In order to introduce stationarity, first the auto-covariance function has to be introduced. The auto-covariance reads

$$\gamma_X(r,s) = Cov(X_r, X_s) = \mathbb{E}[(X_r - \mathbb{E}X_r)(X_s - \mathbb{E}X_s)] \tag{2.57}$$

Given this function a stochastic process is stationarity if

1. $\mathbb{E}[X_t^2] < \infty, \quad \forall t \in \mathbb{Z}$
2. $\mathbb{E}[X_t] = \mu_X, \quad \forall t \in \mathbb{Z}$
3. $\gamma_X(r,s) = \gamma_X(r+t, s+t), \quad \forall r,s,t \in \mathbb{Z}$

This formulation of stationarity is often referred to as weak stationarity [12], in order to contrast strict stationarity. Strict stationarity is a more restrictive property and can be formulated as:

$$f(X_t, X_{t+1}, ..., X_{t+h}) = f(X_s, X_{s+1}, ..., X_{s+h}), \forall s,t,h \in \mathbb{Z}. \tag{2.58}$$

Strict stationarity holds if the distribution is equal for all set length sections of the stochastic process. A strictly stationarity process is also weakly stationarity if the process' second moments are finite [12]. The strict stationarity is, however, difficult to asses from a single observation and more strict than what is needed for most applications [67]. As a result, weak stationarity will often be used instead and will further be referred to as stationarity.

The assumption of stationarity is crucial since with its restrictions also comes some structure. The required regularity of the mean and auto-covariance in the stationarity allows for estimation of these by averaging [67]. This can be clearly seen with the auto-covariance function, which can be restated as

$$\gamma_X(h) = Cov(X_r, X_{r+h}), \quad \forall r \tag{2.59}$$

if the process is stationary.

2.9.2 Autocorrelation

The auto-correlation is simply the correlation between two points in a series. As a result of the simplification in the auto-covariance for stationary processes, see Equation 2.59, the autocorrelation function (ACF) of a stationary time-series reads:

$$\rho_X(h) = \frac{Cov(X_t, X_{t+h})}{\sqrt{Var(X_t) \cdot Var(X_{t+h})}} = \frac{Cov(X_t, X_{t+h})}{Var(X_t)} = \frac{\gamma(h)}{\gamma(0)} \tag{2.60}$$

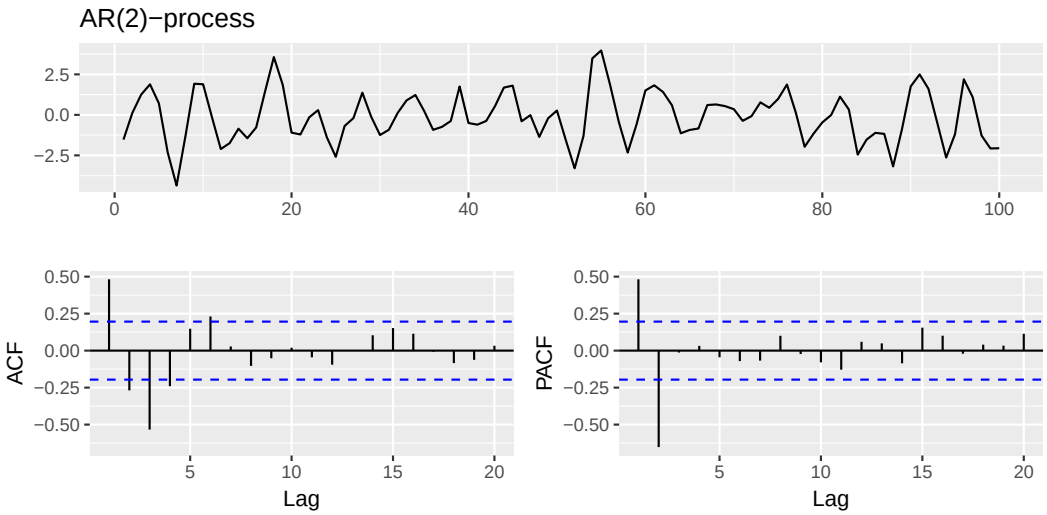


Figure 2.9: An observation of a stationary AR(2)-process with belonging ACF and PACF at different lags.

The autocorrelation captures important properties of the time-series, but is not able to measure conditional dependence. Assume, for example, a process $X_t = 0.8X_{t-1} + \varepsilon_t$, with $\varepsilon_t \sim \mathcal{N}(0, \sigma^2)$. The autocorrelation for lag 1 will then be $\rho_X(1) = 0.8^2$ and by propagation, the higher autocorrelations will also be positive; for example $\rho(2) = \rho(1)^2 = 0.64$. In order to account for such effects, the partial autocorrelation function (PACF)

$$\pi(h) = Cor(X_t, X_{t+h} | X_{t+1}, ..., X_{t+h-1}) \tag{2.61}$$

is introduced and is the autocorrelation between two points in a series after first taking the intermediate points into account. In the previous example, the partial autocorrelation at $h = 2$ equals zero. In order to use these in practice, the theoretical ACF and PACF will have to be estimated. As these depend on the auto-covariance, they can be estimated by averaging for stationary processes. The sample or empirical auto-covariance is defined as:

$$\hat{\gamma}(h) = \frac{1}{n} \sum_{i=1}^{n-h} (x_i - \bar{x})(x_{i+h} - \bar{x}) \tag{2.62}$$

with $\hat{\gamma}(-h) = \hat{\gamma}(h)$ and $\bar{x}$ the sample mean[3]. The divisor is n rather than $n - h$ in order to keep the estimated covariance matrix positive semi-definite [12]; a condition for a covariance matrix of a stationary process. With the estimated covariance the empirical autocorrelation simply reads:

$$\hat{\rho}(h) = \frac{\hat{\gamma}(h)}{\hat{\gamma}(0)} \tag{2.63}$$

The sample autocorrelation and partial autocorrelation are very important tools for investigating dependence structures in a stationary time-series [26]. An example of a stationary process and empirical ACF and PACF at different lags can be seen in Figure 2.9.

2.9.3 Stationary linear processes

The linear stationary processes constitutes a very important class of processes that plays a key role in modeling of time-series [12]. This includes $ARMA$-processes, which is the focus of

[2] $Cor(X_t, X_{t-1}) = \frac{Cov(0.8X_{t-1}, X_{t-1}) + Cov(e_t, X_{t-1})}{\gamma(0)} = 0.8\frac{\gamma(0)}{\gamma(0)} = 0.8$

[3] $\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$

this section. As a bonus, the linear nature of the $ARMA$-process leads to simple methods for forecasting, which makes it convenient in prediction of time-series.

2.9.3.1 AR-processes

A simple class of linear processes are the autoregressive (AR) class of processes. The key idea of the autoregressive model is that a value in a series can be explained by a function of former values [67]. A zero mean AR-process of order p, aptly named an $AR(p)$-process, can be stated as follows

$$X_t = \phi_1 X_{t-1} + ... + \phi_p X_{t-p} + e_t = \sum_{i=1}^{p} \phi_i X_{t-i} + e_t, \quad e_t \overset{iid}{\sim} f(e). \tag{2.64}$$

with the errors e_t following an arbitrary zero-mean distribution $f(e)$. We can without loss of generality assume that the mean of a process is zero as we can simply construct a series $X'_t = X_t - \mathbb{E}[X_t]$, which has zero mean. In order to make notation more convenient we introduce the **back-shift operator** $BX_t = X_{t-1}$, also called the lag operator. The operator is defined in such a was that $B^k X_t = B^{k-1} X_{t-1} = X_{t-k}$. Now using the back-shift operator we can define an AR process in terms of its *characteristic polynomial* $\Phi(B) = 1 - \phi_1 B - \phi_2 B^2 - ... - \phi_p B^p$:

$$\Phi(B) X_t = e_t \tag{2.65}$$

This characteristic polynomial, or AR-operator, contains information about the AR-process. For example, the roots of it has to lay outside of the unit-circle in order for the process to be stationary whereas the opposite is true if they lie inside [10].

An $AR(p)$-process has an interesting relationship with the ACF and PACF. The autocorrelation of an $AR(p)$-process is infinite in extent, whereas the partial autocorrelation is zero for lags larger than p [10]. This makes the PACF very useful in determining the order of an $AR(p)$-process, whereas the ACF is less so.

2.9.3.2 MA-processes

The moving-average (MA) scheme takes another approach and X_t will instead be dependent on previous errors rather than previous observations. This could for example be convenient in a model when shocks in a market increases or decreases the sales of the next period. A moving-average of order q, $MA(q)$, can be stated as:

$$X_t = \theta_1 e_{t-1} + ... + \theta_q e_{t-q} + e_t = \sum_{i=1}^{p} \theta_i e_{t-i} + e_t, \quad e_t \overset{iid}{\sim} f(e) \tag{2.66}$$

Just like in the autoregressive case, an MA-process has a characteristic polynomial $\Theta(B) = 1 + \theta_1 B + ... + \theta_q B^q$ and can therefore be formulated as $X_t = \Theta(B) e_t$. By being a finite combination of i.i.d. errors, an $MA(q)$-process is always stationary [12]. However in contrast to an $AR(p)$-process, the true ACF of an $MA(q)$-process is zero for lags larger than q and will be different from zero at lag q [67]. As a result, the ACF is very useful in determining an order of an MA process.

2.9.3.3 ARMA-processes

The AR- and MA-processes both have useful properties and can be combined in in an ARMA-process. The formulation of such a process can be stated as seen in Equation 2.67.

$$X_t = \sum_{i=1}^{p} \phi_i X_{t-i} + \sum_{i=1}^{p} \omega_i e_{t-i} + e_t. \tag{2.67}$$

By using the characteristic polynomials of both the AR and the MA process this can equivalently be stated as below.

$$\Phi(B) X_t = \Theta(B) e_t. \tag{2.68}$$

The previous conditions on the characteristic polynomial for the $AR(p)$ part of the process still have to be fulfilled in order for the $ARMA(p,q)$-process to be stationary. Rather than cutting of for both the ACF and the PACF will tail of for an $ARMA(p,q)$ process [67].

2.9.3.4 Forecasting

Forecasting is the estimation of future values of a time-series. The principle is to attempt to forecast predictable behaviour by taking the previous information into account [12]. While there are many ways to forecast depending on wanted results, a common description of the "best" forecast is to minimise the mean squared error of the forecasts, cf. [12], [10] and [26]. The estimator minimising the mean squared error of a forecast is the expectation conditioned on the previous observations $\mathbb{E}[X_{n+k}|X_{1:n} = x_{1:n}]$ [67]. This is makes forecasting $AR(p)$ processes simple as the expectation one step ahead in time looks like as follows.

$$\begin{aligned}\mathbb{E}[X_{t+1}|X_{1:t} = x_{1:t}] &= \mathbb{E}[\phi_1 X_t + \phi_2 X_{t-1} + ... + \phi_p X_{t-p+1} + e_{t+1}|X_{1:t} = x_{1:t}] \\ &= \phi_1 x_t + \phi_2 x_{t-1} + ... + \phi_p x_{t-p+1} + \underbrace{\mathbb{E}[e_{t+1}|X_{1:t}]}_{=0}\end{aligned} \tag{2.69}$$

We call a forecast of l steps ahead in time a lead l forecast. The estimation of a lead l forecast of a point x_k can be recursively estimated through the method in Equation 2.69, resulting in the following formulation:

$$\hat{x}_k^{(l)} = \sum_{i=1}^{p} \phi_i \tilde{x}_{k-i}, \quad \tilde{x}_{k-i} = \begin{cases} x_{k-i}, & \text{if } i \geq l \\ \hat{x}_{k_i}^{(l-i)}, & \text{otherwise} \end{cases} \tag{2.70}$$

An $MA(q)$-process is not as straight forward to forecast as an $AR(p)$-process since there are no realizations of the errors available. As a result the errors also have to be estimated. This can be done through invertability, if the $MA(q)$-processes has the property, and given these estimates a lead l forecast of an $MA(q)$-processes can be stated as:

$$\hat{x}_k^{(l)} = \sum_{i=l}^{q} \theta_i \hat{e}_{k-i} \tag{2.71}$$

For details see appendix A.3.

The forecast of an $ARMA(p,q)$-process is simply a combination of the $AR(p)$-process and $MA(q)$-process forecasts. A formulation of a lead l forecast can be seen in Equation 2.72.

$$\hat{x}_k^{(l)} = \sum_{i=1}^{p} \phi_i \tilde{x}_i + \sum_{j=l}^{q} \theta_j \hat{e}_j. \tag{2.72}$$

There are multiple ways of stating the forecast for an $ARMA(p,q)$-process, but this is the most convenient for routine calculation [10].

2.9.3.5 Estimation

So far it has been assumed that the parameters ϕ and θ are known. This is in general unrealistic and the parameters have to be estimated from the realization of the stochastic process. There are multiple different schemes for the estimation, some are only applicable to $AR(p)$-processes, however, two common methods that work for full $ARMA(p,q)$-processes are the maximum likelihood and the least squares methods. The least squares method, once again, relies on minimising the sum of squared errors. The maximum likelihood is a more difficult problem, but has the benefit of using all available information rather than just the first and second moments [26]. Just like in the case of regression the MLE depends on a likelihood function and a joint probability distribution therefore has to be assumed. The optimisation is, however, not as straight forward and the solution to the MLE will in general be found through numerical methods, such as the Newton-Raphson method [67].

2.9.4 ARMA-model selection

The important task of order-selection is a part of the larger encompassing problem of model-selection. If the orders p and q were known in advance it would be straight forward to estimate an $ARMA(p,q)$-process, but since this is rarely the case, the orders will also have to be appropriately chosen [12]. There are multiple tools which can be used in order to perform this task, including visual inspection and model criteria. The visual inspection method uses the properties of the ACF and PACF for the different linear processes; a summary can be seen in Table 2.1. These can be used as a means of identifying both q and p.

Table 2.1: Behavior of ACF and PACF for ARMA models[4]

	AR(p)	MA(q)	ARMA(p,q)
ACF	Tails off	Cuts off after lag q	Tails off
PACF	Cuts off after lag p	Tails off	Tails off

The visual inspection can however in some situations be hard to interpret. While the order identification of pure $AR(p)$ and $MA(q)$ processes is rather simple, the ACF and PACF of an $ARMA(p,q)$-process with non-zero q and p is more difficult to interpret [12]. In order to help with decisions in such cases, model criteria can prove useful. The Akaike information criterion (AIC), formally introduced by Akaike [1] in 1974, is a way of penalizing the model complexity by the amount of parameters within a model. The method adds a term onto the negative log-likelihood and reads:

$$\mathrm{AIC}(\theta) = -2 \cdot \ell(\theta; y) + 2k \tag{2.73}$$

where k is the amount of non-zero parameters. This can then in turn be minimized in order to find an optimal model according to the criterion. Due to the discrete nature of the criterion, this will however have to be done fitting a model multiple times with a different set of coefficients chosen to be zero. In a time-series context this can done by fitting models of different orders, say orders belonging to the set $\{(p,q) : p = 0,...,k,\ q = 0,...,k\}$ for example. Akaike suggests the use of the method to determine order of autoregressive order, but the criterion can be used to determine the orders of an $ARMA(p,q)$-process as well. These two methods can be used in conjunction to increase confidence in results.

2.9.5 Trend and Seasonality

There are other time dependent components that could be present in a series. The monthly average sales might for example show a clear decrease or increase specific seasons, or the sales might be increasing with time. The first is called a seasonal trend, or seasonality, whereas the latter is called a trend. It is clear that if any of these are present in the time-series it cannot be stationary since the unconditional mean will be non-constant. These will therefore have to be removed in order to perform stationary time-series analysis.

The trend and seasonality components are usually identified through visual inspection of the time-series but are also, to some extent, described by the autocorrelation [12]. Much like with errors, two common models are the structures are additive $y_t = x_t + T_t + S_t$ and multiplicative models $y_t = x_t \cdot T_t \cdot S_t$ which can be linearized with transformations [46].

Two methods for incorporation of trend and seasonality is differencing and estimation of the components and focus will be on the latter. Differencing tries to eliminate trend and seasonality by restating the series $x_t = y_t - y_{t-t_S}$, where t_S is the time of the seasonality; for example $t_S = 12$ for monthly data. The regression alternative uses regression techniques for estimation of some assumed seasonal structure. Cryer and Chan [26] gives the examples of using seasonal means, a mean for each season, or cosine based approaches for the predictors of such a regression.

[4]Table re-illustrated from Shumway and Stoffer [67]

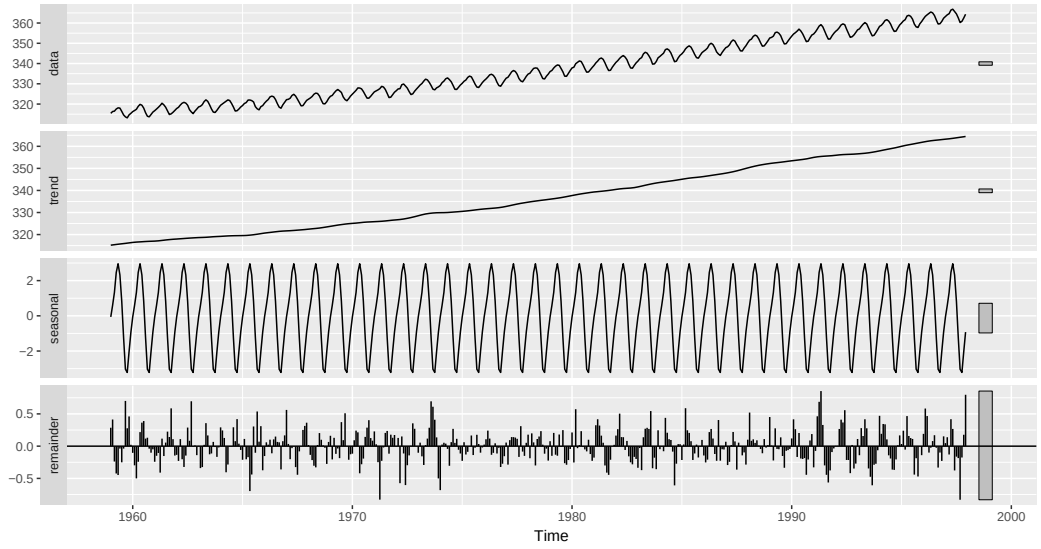


Figure 2.10: An STL decomposition of the atmospheric CO2 concentration data measured at Mauna Loa Observatory from 1959 to 1997.[5]

The Seasonal and trend decomposition using LOESS (STL) method, introduced by Cleveland et. al. [23], takes a similar approach instead using LOESS to estimate relationships. The locally estimated scatter-plot smoothing (LOESS) is a robust non-parametric local polynomial regression able to capture non-linear relationships, see [24]. By stepwise estimating the trend and seasonality and smoothing with LOESS, the STL manages to be versatile and robust [46]. An example of an STL decomposition can be seen in Figure 2.10.

2.9.5.1 Trend and seasonality in MMM

Trend and seasonality are issues previously addressed in MMM, for example see [25], as different seasonal patterns exists for sales within different industries and that companies invest in seasonal patterns [41]. Hanssens et al. [41] claims that there is a danger of using seasonally-adjusted data, and that one should rather treat the seasonality and trend components as an integral part of the model. This, since the seasonality in one predictor can be caused by the seasonality in other predictors. For example, many companies spend a lot of resources on marketing around holidays, such as Christmas, creating a seasonal pattern in the media spend predictors which affects the sales.

2.9.6 Regression with auto-correlated errors

When performing a regression analysis, the possibility of auto-correlated errors should always be considered [10]. As stated in Section 2.2.3, if the errors are correlated the OLS estimator will no longer retain the BLUE property. Instead, when $Cov(\varepsilon) = \Sigma\sigma^2 \neq \mathbb{I}\sigma^2$, the generalised least squares (GLS) estimator $\hat{\beta}^{GLS} = (X^T\Sigma^{-1}X)^{-1}X^T\Sigma^{-1}y$ is the best linear unbiased estimator in the sense of estimator variance [10]. Once again, the GLS estimator is also equivalent to the maximum likelihood estimator of a normally distributed regression: $y \sim \mathcal{N}(X\beta, \Sigma\sigma^2)$. The difficulty lies in estimating Σ. This estimation is not possible without an assumed structure on the correlation as Σ contains $(n^2+1)/2$ unique, unknown elements to be estimated with n equations. One strategy to do this is to assume an $ARMA(p,q)$-process on the errors. Given

[5] Data from Whorf et. al. [60]

a regression function g, this turns out to be

$$Y_t = g(X) + \varepsilon_t, \quad Var(\varepsilon) = \Sigma \tag{2.74}$$

where the errors ε_t follow an $ARMA(p,q)$-process. Now, assume the covariance matrix is positive definite. It is then possible to with a decomposition $CC^T = \Sigma^{-1} \Rightarrow \Sigma = (C^{-1})^T C^{-1}$ restate the problem as

$$CY = Cg(X) + C\varepsilon, \quad Var(C\varepsilon) = C[Var(\varepsilon)]C^T = C\Sigma C^T = \mathbb{I} \tag{2.75}$$

Further, in a linear setting this becomes an ordinary linear model with uncorrelated errors.

$$\tilde{Y} = \tilde{X}\beta + \tilde{\varepsilon}, \quad \tilde{Y} = CY,\ \tilde{X} = CX,\ \tilde{\varepsilon} = C\varepsilon \tag{2.76}$$

The core idea is to then find the covariance matrix Σ from an estimation of the $ARMA(p,q)$-process by estimating the coefficients and then using the autocorrelation function. A simultaneous optimization of both β and parameters belonging to the $ARMA(p,q)$-process can be estimated simultaneously given an order p and q. Box et. al. [10] suggest to fit an OLS and estimating the order from the residuals and then validating if the residuals of the GLS are uncorrelated.

2.9.6.1 Forecasting in a regression model with auto-correlated errors

Forecasting can be easily done when realisations of predictors are known [10]. This can for example be the case when dummy predictors are used to model seasonality. The lead l forecast of y_k is

$$\hat{y}_k^{(l)} = x_k^T \hat{\beta} + \hat{\varepsilon}_k^{(l)} \tag{2.77}$$

where $\hat{\varepsilon}_k^{(l)}$ is the lead l estimate of the $ARMA(p,q)$-process.

2.10 Empirical model evaluation

The formal theory of comparing estimators is called decision theory [73] and introduces the concept of a *loss function*, sometimes called *cost function*. This loss function $L(Y, g(X))$ penalizes errors in predictions and leads to a criterion of choosing an estimator $\hat{f}$; the expected prediction error $\mathbb{E}[L(Y, g(X))]$ [44]. The $\mathbb{L}_2$ loss function, $L(Y, f(X)) = (Y - g(X))^2$, is an example of such a loss function. With realized data the expected prediction error can then be approximated with

$$\frac{1}{n}\sum_{i=1}^{n} L(y_i, g(x_i)) \approx \mathbb{E}[L(Y, g(X))] \tag{2.78}$$

if n is large enough. Inserting the $\mathbb{L}_2$ loss function into Equation 2.78 yields

$$L^{RSS}(y, G(X)) = \sum_{i=1}^{n} (y_i - g(x_i))^2 \tag{2.79}$$

which is the previously described residual sum of squares. This cost function can be stated in any way suitable. For example, if the observations are i.i.d. the maximum likelihood can be stated as a cost-function through the log-likelihood, seen below.

$$L(Y, g(X)) = -\ell(\theta; Y, X) \Rightarrow L^{MLE}(y, G(X)) = -\sum_{i=1}^{n} \ell(\theta; y_i, x_i)$$

In the majority of cases, the interest in the modelling is not to find the function that best fits the data we have, but one that generalises well [3]. The expected error of a prediction outside of the set of realised values a model was fit on is called the **generalisation error**, or

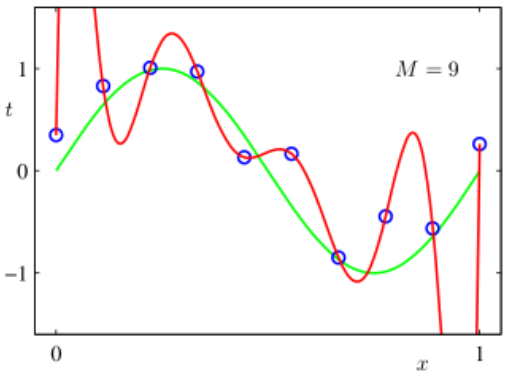


Figure 2.11: The red line represents an overfitted model, in comparison to the green line with a better generalisation performance.

Source: Bishop [6]

test error [44]. Given the realised values $\mathcal{D} = (\mathbf{x}_1, y_1), .., (\mathbf{x}_n, y_n)$ and estimator $\hat{g}$ and a loss function L, the generalisation error reads

$$\mathbb{E}[L(Y, \hat{g}(X))|D],$$

with X and Y randomly drawn from their joint probability distribution. In practice, assessment of the generalisation performance is immensely important as it helps in the choice of learning methods or models [44]. The true generalisation error is however a theoretical measure and needs to be estimated. This turns out to be difficult and most methods instead estimate the aforementioned expected prediction error (or expected test error) $\mathbb{E}[\mathbb{E}[L(Y, \hat{g}(X))|D]]$ [44].

If a model does not generalise well, the model might be so called overfitting to the data used for training the model, yielding a poor generalisation performance (see Figure 2.11).

2.10.1 Bias-variance tradeoff

The bias-variance tradeoff is a central element of modern machine learning and statistical methods that should always be addressed and discussed. Since the estimator is a function of random variables the estimator itself will also be a random variable. As such, there is uncertainty in the estimation represented by a distribution. The core idea of the bias-variance tradeoff is that restrictions can be put on an estimator in order to reduce the variance of the estimator. The side effect, however, is often introduced systematic error made by the estimator, called bias.

Definition 11 (Bias)**.** *Let $\hat{\theta}$ be an estimator of θ, then*

$$bias(\hat{\theta}) = \mathbb{E}_\theta[\hat{\theta}] - \theta$$

is the bias of the estimator.

Using this definition we call an estimator $\hat{\theta}$ of θ unbiased if $bias(\hat{\theta}) = 0$; the estimator is correct on average. Unbiased estimators used to be considered more important but their considered importance has decreased [73]. Instead the accuracy of the estimator is in focus.

Usually, increasing the model complexity results in a lower bias and a higher variance [44]. An example of such a relation can be seen in Figure 2.12. Reducing the model complexity can be done in multiple ways and in regression, shrinkage methods and variable selection are two approaches.

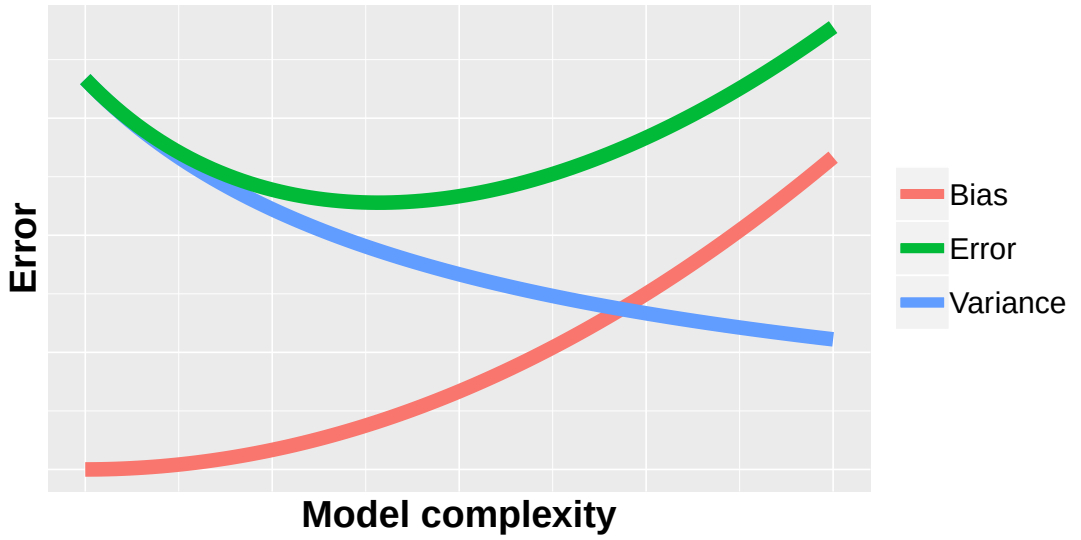


Figure 2.12: Example of changes in variance and error with increasing bias.

2.10.2 Test, train and validation sets

One way to measure the generalisation performance is to split the set of data into a training and test set. This is known as the simple hold-out method as described by Arlot et al. [2]. The data is trained on the training set to later be tested and validated on the test set, from which the generalisation error is obtained. It is an estimation of the true generalisation error since the true generalization error is hard to obtain. However, if many iterations of the model design is performed over a limited amount of data, over-fitting to the validation set can occur and it may be necessary to have a third test set on which the final evaluation of selected models take place [6]. A general rule of how this split is made is difficult to formulate since it depends on the signal-to-noise ratio [44].

2.10.3 Cross-validation

In many cases, such as in MMM, available data is limited and as much of it as possible have to be used in order to create models with good generalization. However, if the validation set is small it will yield an unprecise estimate of the prediction error [6]. One solution to this is cross-validation (CV). Here, the data is first split up in a training and test set as in the hold-out method, seen above. The training set is then split in K equally, or close to equally, large partitions. The model is then fit K times, each time with one of the partitions as a validation set and all other as a training set. This process is specifically called K-folds cross-validation. The validation-loss is then averaged over the validation sets resulting in a final generalisation error. A visualization of the procedure can be seen in Figure 2.13.

A special case of the K-folds CV, called leave-one-out CV, is when number of partitions equals number of data points, $K = n$. The estimator is then, for every data point, fit on all but one data point which serves as validation. This increases the times the model is fit to K, which can be problematic if the training is computationally expensive.

Since MMM is a time-series oriented problem as points are distributed on a weekly basis, other methods taking the time-dependency into account might be suitable. One such method is described in Algorithm 3. Arlot et al. [2] stresses the importance of validating the model in this way in problems with time series, since data points in time series data are dependent on each other. Thus, each point is then predicted, and the model is later fitted with the new

[6] Inspired by Hyndman and Athanasopoulos [46].

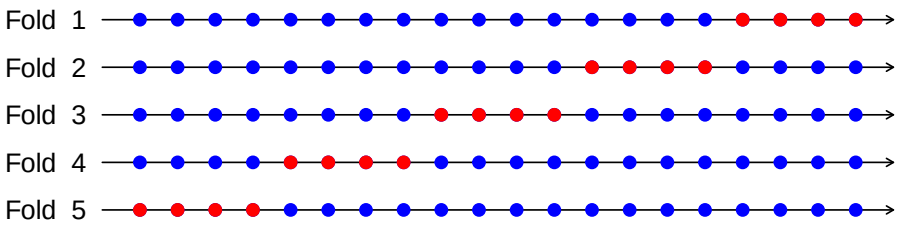


Figure 2.13: Visualization of a 5-fold cross validation. Blue points represents the training set and red the validation set for each fold [6].

Algorithm 3 The alternative validation method as used commonly in time-series.

procedure TIME-SERIES CROSS-VALIDATION
  Given model $\mathcal{A}$ data $\mathcal{D} = \{(x_1, y_1), .., (x_n, y_n)\}$, training set fraction $r \in]0, 1[$
  Divide data into training set $I^{(t)}, |I^{(t)}| = \lceil |\mathcal{D}| \cdot r \rceil$ and validation set $I^{(\nu)} = D \setminus I^{(t)}$
  for i in $1..|I^{(\nu)}|$ **do**
    Fit model $\mathcal{A}$ to $I^{(t)}$
    Predict new point $\hat{y}_i = \mathcal{A}(x_i | I^{(t)})$
    Set $I^{(t)} = I^{(t)} \cup \{(x_i, y_i)\}$
  end for
  Obtain generalisation error $\hat{\mathcal{L}}\left(\mathcal{A}; D; I^{(t)}\right) := \frac{1}{n_\nu} \sum_{i \in I^{(\nu)}} \gamma\left(\mathcal{A}(I^{(t)}); I^{(\nu)}\right)$
end procedure

point for next week, and so on. The generalisation error is then estimated based on these predictions. A visualization of the procedure can be seen in Figure 2.14.

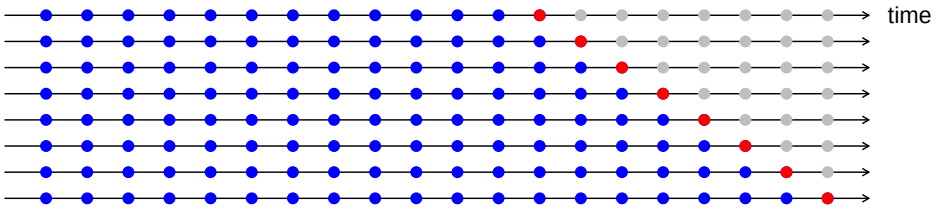


Figure 2.14: Visualization of a time series cross validation. Blue points represents the training set and red the validation set for each fold[7].

2.10.4 Bootstrap

In order to measure the certainty and not the accuracy of an estimate, one cannot simply rely the generalisation error. A model might display very accurate estimates with a low generalisation error, but it can also be relevant to know the certainty of specific statistics of a model, such as coefficient estimates. In parametric MMM models, it is crucial to be confident about the media coefficient estimates, which brings us to methods measuring this.

Bootstrap is a re-sampling method proposed by Efron [30] and is a general tool for assessment of statistics over the data. The method has many uses such as estimation of confidence intervals and standard deviation. It can also be used to evaluate accuracy of the model. How-

[7] Inspired by Hyndman and Athanasopoulos [46].

ever, consistent with cross-validation, the bootstrap tries to estimate the generalisation error but usually only estimates the prediction error well [44].

Let $Z \sim F_Z$ be a random variable, distributed after an unknown distribution F_Z, and a set of realised values be denoted by $\mathbf{z} = \{z_i : i = 1, ..., n\}, z_i = (x_i, y_i)$. Then given a random variable $T(Z, F_Z)$ which might depend on both Z and F_Z, the goal of the bootstrap is to estimate the sampling distribution of T from z [30]. This is done through the construction of a sample probability distribution $\hat{F}_Z$ which is the uniform distribution over $\mathbf{z}$; equivalently the distribution with mass $1/n$ at each point $z_1, ..., z_n$. The bootstrap distribution of $T(Z^*, \hat{F}_Z)$, with $Z^* \sim \hat{F}_Z$, is then equal to the distribution of $T(Z, F_Z)$ if $\hat{F}_Z = F$. How well this approximation works depends on the choice of T [30].

Calculation of the bootstrap distribution F_{T^*} is the difficult part, and while methods such as direct computation can be possible, a Monte Carlo approximation is always applicable given a realised set of data. This approximation works through repeated realisations of Z^* by sampling the distribution $\hat{F}_Z$ resulting in samples of size n: $\mathbf{z}^{*(1)}, ..., \mathbf{z}^{*(B)}$, for some large B. The samples $T(\mathbf{Z}^{*(b)}, \hat{F}_Z),\ b = 1, ..., B$ are then used to construct an empirical cumulative distribution function (cdf) F_{T^*}. An example of the Monte Carlo bootstrap procedure can be seen in Algorithm 4.

Algorithm 4 Bootstrap

Let $Z \leftarrow (\mathbf{x}_1, y_1), ..., (\mathbf{x}_n, y_n)$ and T_n be some statistic to be evaluated.
for $b \leftarrow 1$ to B **do**
    for $i \leftarrow 1$ to n **do**
        $Z_i^{*(b)} \sim \text{Uniform}(Z)$
    end for
    $T_n^{*(b)} = T_n(Z^{*(b)})$
end for
$F_{T_n^*}(\cdot) = \frac{1}{B} \sum_{b=1}^{B} \mathbb{1}_{\{T^{*(b)} \leq \cdot\}}$

The statistic to be evaluated can be chosen out of a wide range, for example standard deviations and confidence intervals for estimated parameters. There are multiple methods for how the confidence intervals can be estimated from a bootstrap distribution, with the percentile method being one of the simpler. The percentile method bases its confidence interval estimation solely on the (empirical) bootstrap cdf

$$F_{T^*}(\cdot) = P_*(T^* \leq \cdot) = \frac{\#\{T^{*(b)} \leq \cdot\}}{B}.$$

It approximates the quantiles of the true cdf directly from the empirical cdf as follows. Let $\delta_P(\alpha) = F_{T^*}^{-1}(\alpha)$ denote the α-quantile of the cdf. Then the $(1 - 2\alpha)$-percentile confidence interval is received with $[\delta_P(\alpha), \delta_P(1 - \alpha)]$. The percentile method's main advantage is its simplicity.

As an extension of the percentile method, the bias corrected (BC) and bias corrected and accelerated (BCa) confidence intervals have been introduced by Efron [34, 33], with the BCa being an extension of BC. The BC method assumes that there exists a monotone function g with a bias-constant z_0 such that $\phi = g(\theta)$ and $\hat{\phi} = g(\hat{\theta})$ with

$$\frac{\hat{\phi} - \phi}{\tau} \sim \mathcal{N}(-z_0,\ 1), z_0, \tau \text{ constant}.$$

Given these assumptions, the correct confidence intervals for ϕ are identifiable and can be converted back to confidence intervals for θ. The BC method manages to do this automatically without any need to specify a function g [33]. The assumptions might seem quite restrictive. However, this method is actually a more general version of the percentile method which is only correct if, under similar assumptions, ϕ is distributed as $\hat{\phi} \sim \mathcal{N}(\phi,\ \tau^2)$ [31]. The BC intervals

are specified as follows:

$$\delta_{BC}(\alpha) = F_{T^*}^{-1}\left(\Phi\left\{2\rho_0 + \rho^{(\alpha)}\right\}\right) \tag{2.80}$$

with Φ denoting the cdf of the standard normal distribution, $\rho_o = \Phi^{-1}(F_{T_n^*}(T_n(\mathbf{z})))$ and $\rho^{(\alpha)} = \Phi^{-1}(\alpha)$. These bias-corrected bootstrap intervals are generally different from the real by $\mathcal{O}(n^{-1})$ as $\delta_{BC}(\alpha) = \delta_{true}(\alpha) + \mathcal{O}(n^{-1})$ [29].

The bootstrap is a powerful method. However, due to the i.i.d. assumption of the bootstrap method it usually fails for statistics applied on dependent predictors as it ignores the order of them [14]. To counteract this, blockwise bootstrap is a bootstrap method that includes time-dependencies.

2.10.4.1 Blockwise bootstrap

The blockwise bootstrap method, introduced from a theoretical viewpoint by Künsch [50] in 1989, is an extension of the bootstrap method which allows incorporation of dependent predictors. The method re-samples blocks $X_{t+1}, ..., X_{t+l}$ from a series $\{X_t : t = 1, ..., n\}$ to try to copy the behavior of an estimator $\hat{\theta}$ [13]. This allows the bootstrapping samples to retain time-dependencies which are important for statistics which depend on them; for example the autocorrelation. The blockwise bootstrap is non-parametric and entirely model-free for stationary observations [14], which is one of its main advantages. As a result, the method is robust against misspecifications in the model.

The blockwise bootstrap works as follows. Let $\{X_t : t = 1, ..., n\}$ be a stationary time-series. First, the time-series is partitioned into overlapping blocks S_i, with some length l, which can be represented by the function

$$F(X_1, ..., X_n; l) = S_1, ..., S_{n-l+1}$$
$$S_i = \{X_i, ..., X_{i+l-1}\}, \quad S_{i,j} = X_{i+j-1}.$$

The overlapping blocks S_i are then uniformly sampled with replacement k times, with $k = \lceil \frac{n}{l} \rceil$, to create a set new set of blocks $S^* = \{S_1^*, ..., S_k^*\}$. The sampled blocks are then concatenated to create a new time-series X^* with the same length as the original time-series X. The new time-series will then read

$$X^* = \{S_{1,1}^*, ..., S_{1,l}^*, S_{2,1}^*, ..., S_{2,l}^*, ..., S_{k,1}^*, ...S_{k,h}^*\}, \quad h = n \mod l$$

and has its last block cut of if it exceeds the length of the original time-series X. Just like the regular bootstrap, this sample can then be used to compute the statistic $T_n^* = T_n(X^*)$. Once again, this is done many times in order to get a distribution over the statistic with cdf $F_{T_n^*}$.

The blockwise bootstrap has been justified both theoretically and empirically for multiple statistics. Examples include generalised M-estimators for $AR(p)$-processes, see [15], and estimation of median, see [13]. An especially useful finding is that the blockwise bootstrap has been proven consistent for smooth linear statistics, cf. [50].

In order to estimate confidence intervals, the percentile method can also be used for the dependent case. However, for such statistics the more sophisticated methods get more complicated. Götze and Künsch [39] have, for example, generalised the bias corrected and accelerated intervals for the dependent case, however, the formulation is significantly more complicated compared to the original BCa interval formulation.

One issue with the blockwise bootstrap is its dependence on the choice of l and the choice of l can be rather important [32]. This block length is not easy to choose and different method can be used. Künsch and Bühlmann [16] have, for example, proposed a method for estimating the optimal block-length. Further, it has been proven that the optimal block-length is $l(n) = O(n^{1/3})$ where the constant depends on the statistic evaluated [50]. An alternative to the aforementioned estimation is to simply choose a block-length $n^{1/3}$, which works well in many cases [16], or a block-length based on previous similar experiments.

The largest issue with the blockwise bootstrap, however, is that the dependency structure might not hold at block limits as this dependence is ignored. As a result, the bootstrap sample might not be (conditionally) stationary [14].

3 Method

In this chapter, the method how to perform the work is presented. We introduce which models were investigated and clarify how these experiments were made, on which datasets.

3.1 Pre-study

First, a pre-study on MMM and the areas of interest was made. Based on this, theory regarding these areas was written to determine how to perform the experiments and what to consider in the different models. The focus of the pre-study included basic statistical regression concepts such as assumptions for models, as well as marketing theory on how to build statistical models within the area of marketing.

Elements of Statistical Learning [44], Pattern Recognition and Machine Learning [6], Bayesian Data Analysis [38], Bayesian Statistics and Marketing [66] and Applied Regression Analysis A Research Tool [64] have been the main sources, accompanied by relevant papers found online. For studying the concepts of marketing, Market Response Models: Econometric and Time Series Analysis [41] have been the main source, along with with relevant papers found online. Based on the studies of these, suitable models and methods were chosen for implementation in accordance with the purpose of the thesis.

3.2 Datasets

In order to evaluate our method, two data sets were used: a real-world data set and a simulated set. The full evaluation took place on both, however, the objectives of the evaluation differs to some extent. The main goal of the real-world data set was to evaluate the potency of the proposed methods in a real-world setting but is limited in means to validate findings about uncertainty.

The simulated dataset has two main benefits. First, it allows for control of the structure, true coefficient estimates and ensuring presence of multicollinearity, allowing a valid evaluation of the accuracy of estimated coefficients in the presence of multicollinearity. Secondly, it allows for sampling from the true distribution and the true coefficient uncertainty (distribution over the coefficients) can therefore be computed through repeated sampling and fitting. The simulated data-set is however a simplification of reality and the results cannot fully represent

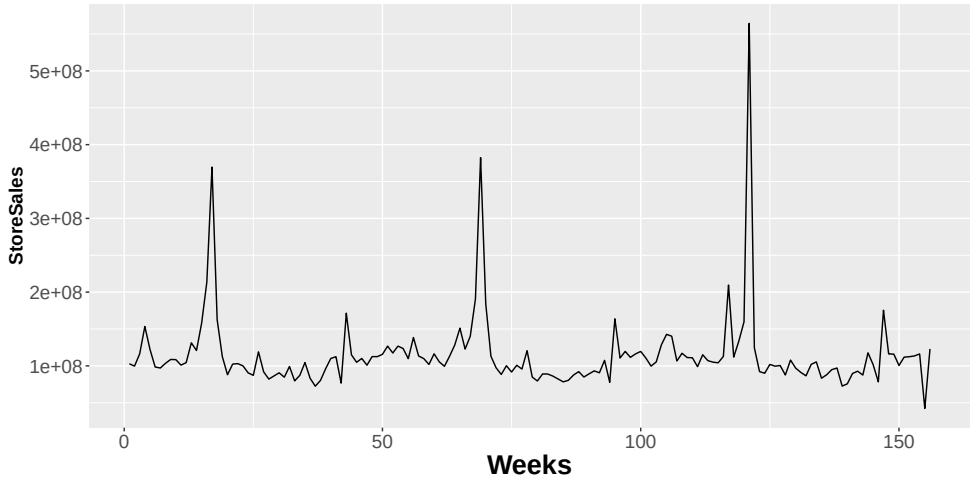


Figure 3.1: The StoreSales predictor real dataset used in this work.

real-world setting. Simulating a data set is a common approach in the field, and has been done in several previous works [48, 69, 53].

3.2.1 Real-world dataset

The real-world data set was provided by Nepa and consist of weekly sales data from a retail company over a period of three years, and the response variable (sales in store) can be seen in Figure 3.1. This consists of sales numbers and marketing channel spending, but also of other control variables with possible relevance, such as temperature, precipitation, CCI and other factors. This data is split up in a couple of ways. First of all, sales, marketing spending and macroeconomic predictors are divided on a country-basis and recorded separately for the nordic countries. Secondly, the sales are split up between store sales and online sales. Lastly, the store sales are present both on a per store basis and summed up, representing the total offline sales in a country. In order to use the data, the following steps were taken.

1. The data was examined and null values were imputed. How these were imputed depended on the structure of the predictor and what appears logical in relation to the predictor's distribution and structure. Although very sophisticated methods for imputing data exists [5], one simple way is to take the mean or the median of the predictor. While this does ignore predictor correlations and creates a bias in form of an underestimation of the variance, this method was chosen in this work due to its simplicity and quick implementation.

2. Once all values had been imputed, the simple decay rate function was applied on the media spend predictors in the data set, as seen in Equation 2.1. The decay rate values λ was set by recommendation from Nepa with its base in examination of previous cases done within similar contexts.

3. Lastly, a subset of predictors were selected for analysis as discussed in 3.3.2. and models were applied on the data set with the method described below.

3.2.2 Simulated dataset

When evaluating the quality of the parameter estimates, it is convenient to know the true structure of the data and the values of the coefficients. Evaluating whether the final contributions of a media channel are correct is easier if one knows the actual contributions of a model.

Functional form	`display...net`	`facebook...net`	`search\_branded...net`
$\mathbf{X}$	24.17	30.20	23.95
$\mathbf{X}_{Log-linear}$	26.84	26.82	25.56

Table 3.2: The VIF-values for the variables in the different functional forms on the simulated data. As can be seen, multicollinearity is strongly suggested, indicating also that the data matrices $\mathbf{X}$ and $\mathbf{X}_{Log-linear}$ are (nearly) singular.

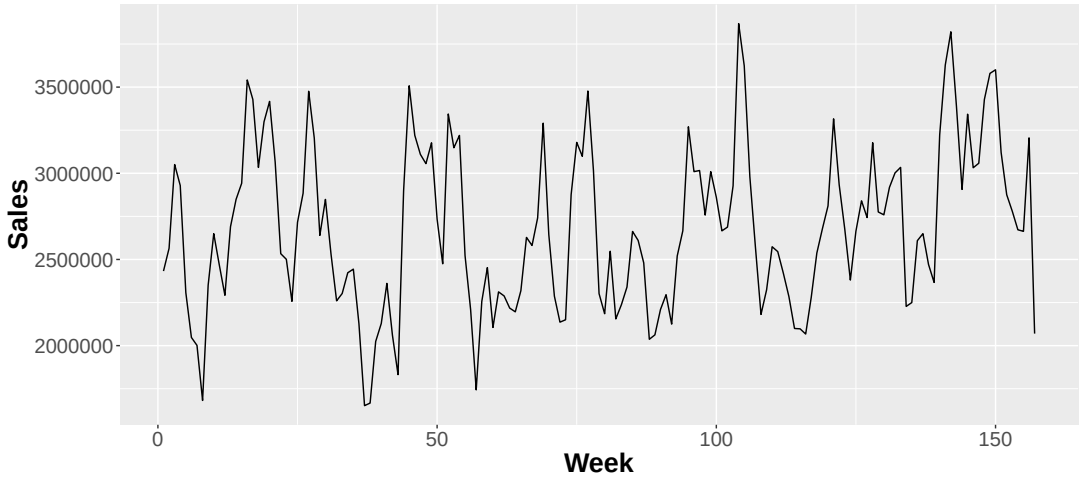


Figure 3.2: The data simulated as described above.

This is the role of the simulated data-set. This simulated data follows the assumptions of a model as seen in 2.52 and is simulated in the following steps. For the investigation to prove valid, there were two important parts to consider. First of all, it was, as mentioned, important to know the true underlying relations between the predictors and the response predictor, rather than how the response predictors are generated. Second of all, as the study tends to examine how well some of the model can mitigate the issue of multicollinearity, multicollinearity was ensured to be present; in particular for the media variables. As seen in Table 3.2, the VIF-values (see Section 2.7.1) for the media spend predictors were all above 20, twice the value (10) that strongly suggests multicollinearity [64].

Table 3.1: Coefficients of the ARMA(2,2)-process used to generate the noise for the simulated data.

	AR	**MA**
1	0.7	0.2
2	-0.3	0.5

1. The predictors were generated using the package `dammmdatagen`[1]. This package mainly uses stochastic processes such as Hidden Markov Models to generate the predictors, which contributes to creating artificial data similar to actual data.

2. The sales data was generated based on an underlying structure

$$\log(y_t) = \beta_0 + \sum_{i=1}^{p} \beta_i \log(x_{i,t}) \forall t \in \{1, .., T\}$$

[1] Available at `https://rdrr.io/github/DoktorMike/dammmdatagen/`. Accessed 18/6 - 2019.

with parameters given. This ensured us to know the true underlying structure of the data and the actual coefficients. This structure of the data was chosen due to Dominique M. Hanssens' argumentation that most empirical evidence points towards the sales response function being concave [41]. The β-coefficients were in turn generated using random generators, sampling them uniformly at random within intervals assumed reasonable after discussion on which ones might be more important and which not. The coefficient values can be found in Appendix B in Table B.1. It is important here to note that the actual values are not very important as the data is generated; the most important is to know the true values of the parameters. Also important to note is that we did not let all predictors affect the result; only 16 were set to be different from 0. This was done since in a real-world setting, all predictors might not have a direct relationship with the actual sales and all data might not be available.

3. When the sales data had been generated, a seasonal component and a trend were added additively as described in the equations below.

$$Seasonality_t = \Big(\frac{1}{T \cdot 100} \sum_{t=1}^{T} Sales_t\Big) \cdot \sin(\frac{2\pi}{26} \cdot Week_t - 10)$$

$$Trend_t = \frac{200 \cdot t}{T \cdot \sum_{t=1}^{T} Trend_t}$$

where $Week_t$ denotes the week of the year of time-step t, and T denotes the total amount of time-steps in the model. These were added on top of the previously generated sales. The equations were chosen based on trials, at the end ending up with data that looks somewhat reasonable. The additive method was chosen for two reasons. First, most methods are based on an additive or multiplicative decomposition [17]. The choice is situation-dependent; if Secondly, it was chosen due to its simplicity.

4. Finally, a time-dependent noise was generated through an ARMA(2,2)-process and added to add noise that the predictors cannot account for. This helped ensuring that there was an auto-correlation between the data points. This ARMA-process was multiplied by $\frac{\sigma_{Sales}}{2}$, where

$$\sigma_{Sales} = \sqrt{\sum_{t=1}^{T} \frac{(Sales_t - Sales_{mean})^2}{T-1}}$$

In other words, the standard deviation of the generated sales. The ARMA-coefficients can be seen table 3.1. This finally yielded the resulting simulated sales response variable, on which the models were applied to. The resulting data can be seen in figure 3.2. It is noteworthy that no outliers were added to the data, and the response variable was later studied and confirmed to not contain outliers, both for the logarithmized response variable and the actual response variable. Here, we define outliers using Tukey's method (see Section 2.7.2). As the real dataset contained about 16 outliers, one would be able to compare the effects of robust regression on the respective data sets.

3.3 Implementation and experiments

The implementation of the experiments was done in R[2] and Python[3], depending on the tools needed and what is available in respective programming language. For example, in Python a pre-made package to compute the SHAP values for a model[4] is available, while it might be more suitable to perform the analysis of the parametric regression models in R due to the availability of statistical packages for this in R.

[2]R project, a language used for statistical modelling. `https://www.r-project.org/`. Accessed 18/6 - 2019.
[3]Programming language used for many applications. `https://www.python.org/`. Accessed 18/6 - 2019.
[4]Package SHAP, available at `https://github.com/slundberg/shap`. Accessed 18/6 - 2019.

3.3.1 Ensuring validity and reliability

To ensure a valid comparison, all models were evaluated using time-series cross-validation on both data sets (see Section 2.10.3). The first 117 data points (75 % of all) were chosen as training points and the predict one, add one scheme was used, as seen in Algorithm 3. These point-wise predictions formed the base for the generalisation error, while the predictions on the training set points provided a comparison between the training and test errors to help determine whether a model is overfitting or not. The errors obtained from the time-series cross-validation was then be used to compare the models between each other in terms of quality of fit.

The scripts developed to perform the experiments followed a clear structure, where the data was first defined in a general way, followed by a section in which each model is evaluated in the same way. The models described in the implementation chapter thus went through the same steps, ensuring a valid comparison.

3.3.2 Feature selection

Although feature selection is an important part of MMM, this part was abstracted in this thesis to reduce the scope. One must however stress that this is an important area that does affect the results significantly. As a feature selection approach cannot be assumed to be perfect, we used 10 out of the 16 predictors that are actually affecting the sales for all models except for the `bsts` model (see section 3.3.5.1). As one wishes to obtain 16 predictors as the recommendation is to have at least 10 points for each estimated weight [43], 6 predictors that are not affecting the sales were also included.

For the real data, predictors were selected based on discussion about previous projects within a similar context. The aim for amount of predictors is about 16 predictors in accordance to the recommendation of 10 data points per estimated weight, for example suggested by Harell. et al. [43].

3.3.3 Trend and seasonality

In order to take out the seasonality and trend component of the data, the package `stl`[5] in R was used. STL decomposes the time series data into trend, seasonality and residual components, taken out using a technique called *Loess* [23].

Due to Hanssens' [41] argumentation that the seasonality should be integrated as a part of the model and not taken out and the fact that many seasonal components does not consider any regressors, the seasonality was taken out on the residuals of an Ordinary Least Squares-model. This seasonality was then reintegrated as an integral part of the model in the form of an additional predictor. Hanssens also means that including dummy predictors for seasons and events like holidays and Christmas can be appropriate to use. Thus, a dummy predictor for Christmas was included. The method used for the trend and seasonality estimation can be described with the following steps:

1. A least squares estimation of only the media predictors is performed with a regular Ordinary-Least Squares regression.

2. A decomposition using STL is performed on the residuals.

3. The seasonal and trend components from the decomposition are added and used as a new predictor for other models.

The method was chosen since it provided the best results in initial tests and serves as a crude aggregation of the trend and seasonality and to some extent allows for simultaneous estimation

[5]Information at `https://www.rdocumentation.org/packages/stats/versions/3.5.2/topics/stl`. Accessed 18/6 - 2019.

of the seasonality, trend and coefficients, albeit with a potentially large introduced error. Trend and seasonality is not examined further as it is not a goal of this thesis.

3.3.4 (Intrinsically) linear models

All models following below were implemented in three functional forms. These include the linear (equation 3.1), the intrinsically linear log-linear (equation 3.2 and 3.3) and the semi-logarithmic and its intrinsically linear form (equation 3.4 and 3.5).

$$y_t = \beta_0 + \sum_{i \in M} \beta_i a_{i,t} + \sum_{i \in \{P \setminus M\}} \beta_i x_{i,t} + \varepsilon_t \tag{3.1}$$

Here, $a_{i,t}$ is defined as the result of the simple decay rate transformation.

$$y_t = \beta_0 \cdot \Big(\prod_{i \in M} a_{i,t}^{\beta_i} \Big) \cdot \Big(\prod_{i \in \{P \setminus M\}} x_{i,t}^{\beta_i} \Big) \cdot \varepsilon_t \tag{3.2}$$

$$\log y_t = \log(\beta_0) + \sum_{i \in M} \beta_i \log(a_{i,t}) + \sum_{i \in \{P \setminus M\}} \beta_i x_{i,t} + \log \varepsilon_t \tag{3.3}$$

$$y_t = \exp\left(\beta_0 + \sum_{i \in M} \beta_i a_{i,t} + \sum_{i \in \{P \setminus M\}} \beta_i x_{i,t} + \varepsilon_t\right) \tag{3.4}$$

$$\log y_t = \beta_0 + \sum_{i \in M} \beta_i a_{i,t} + \sum_{i \in \{P \setminus M\}} \beta_i x_{i,t} + \varepsilon \tag{3.5}$$

Here, M denotes the set of media spend predictors, and P denotes the set of all predictors. Note that for the log-linear functional form (Equation 3.2), a predictor was transformed using a MinMax-scaler between 0 and 1 instead of being log-transformed if it had negative values to be used in the intrinsically linear regression (Equation 3.3). This since negative values cannot be transformed to a logarithmic form.

While none of the models above provide an S-shaped curve as several sources claim is the true response curve for media spend [48, 69, 70], this was not chosen. The main reason for this is due to Hanssens [41] claim that there is no empirical evidence for the S-shaped curve, and rather a concave response curve.

As discussed in section 2.8.2, the linear functional form can serve as a good approximation for the estimation, while the log-linear provides a diminishing returns to scale effect, and the semi-logarithmic model provides an increasing returns to scale which is convenient due to the synergy effect spending in different marketing channels tends to have. These functional forms of parametric models were thus chosen.

3.3.4.1 Ordinary Least Squares

The Ordinary Least Squares model (see Section 2.2.2) was implemented, serving as a baseline to compare the effects of the other models. For this, the `lm` function from the package `stats` in R was used.

3.3.4.2 Generalised least squares

Possible time-dependencies within the model were dealt with through the generalised least squares approach presented in section 2.9. Specifically, the following steps were used.

1. The model was fit according to the least squares estimator.
2. The residuals of this fit were inspected for stationarity and correlation.

3. If correlation was present and stationarity was not rejected, orders p and q were determined by a combination of Akaike's Information Criterion (AIC) (see [44, p. 230]) and visual inspection of the (partial) autocorrelations. If stationarity was rejected or if the series seemed stationary and was uncorrelated, nothing further was done and modelling for the specific model was not performed.

4. Given the determined order a linear model with $ARMA(p, q)$ correlated errors was fit using the function `gls` from the package `nlme` in R[6].

5. The errors were inspected again for autocorrelation and stationarity and if correlation still existed the order of the process was revised.

3.3.4.3 Robust regression

In order to account for uncertainty in terms of outliers and heavy-tailed distributions, robust methods as presented in Section 2.7.2 were evaluated. Although there has not been any evidence found supporting the existence of heavy-tailed error distributions in MMM, the occurrence of outliers is well-known due to markets' unpredictability, as discussed in section 2.8.1.

When implementing robust regression, the function `rlm` from the package `MASS` in R was used, in which both the M-method as proposed by Peter J. Huber [45] and the MM-method can be used. As the MM-estimate has empirically shown superior results to M-estimates, for example by Yu et al. [78], this method was used.

3.3.4.4 Shapley value regression

To try and account for uncertainty in parameter estimates, with its root in multicollinearity, Shapley value regression was implemented (see Section 2.7.1). It was examined whether Shapley value regression could reduce the uncertainty of the parameter estimates, and how this would affect the generalisation error. To implement Shapley Value Regression, the relative importance measures was calculated using the `calc.relimp` function in the `relaimpo`[7] package. In addition, the media coefficients were also forced to be positive in this Shapley Value Regression, therefore introducing a further bias.

3.3.4.5 Constrained regression

In the MMM context, the assumption that media spending has a positive effect can be made and this positive effect is a desired property. In order to incorporate this assumption into a model, a constrained regression model was implemented. By enforcing constraints, such that the solution space is convex, on a convex problem, the new problem will also be convex and can be solved through convex optimisation [11].

To fit a constrained regression model, the package `colf`[8] was used. This package provides a framework to implement linear regression models with constraints, solving the regression as a constrained linear optimisation problem using the gradient based Gauss-Newton algorithm described in section 2.3.2.1.

The coefficients of the media spend predictors were constrained to be larger or equal to 10^{-10} while the intercept, accounting for the base sales, was constrained to be simply positive for all functional forms. 10^{-10} was chosen as media coefficients by Nepa's experience usually are in the magnitude of 10^{-8} in the semi-logarithmic functional form (Equation 3.4). While

[6] Documentation available at `https://cran.r-project.org/web/packages/nlme/nlme.pdf`. Accessed 18/6 - 2019.

[7] Documentation available at `https://cran.r-project.org/web/packages/relaimpo/relaimpo.pdf`. Accessed 18/6 - 2019.

[8] `https://cran.r-project.org/web/packages/colf/colf.pdf`. Accessed 18/6 - 2019.

this differs between different functional forms, this was considered a sufficiently small order of magnitude to not exclude any reasonable values for any functional form.

$$\begin{aligned} \beta_0 &\geq 0 \\ \beta_j &\geq 10^{-10}, \quad \forall j \in M \end{aligned} \tag{3.6}$$

The effects of this constrained model were then evaluated on the same grounds as the normal models.

3.3.5 Bayesian models

Two Bayesian models were implemented: a non-hierarchical and a hierarchical model. MCMC sampling was performed to arrive at distributions for each parameter in the model, which helped determine the confidence intervals for the parameters and determine the quality of the models. For both models, pre-made packages with built-in MCMC samplers were used. Below follows the two models that were implemented.

3.3.5.1 Bayesian Structural Time Series model

The Bayesian Structural Time Series model was chosen as it provides advantages such as including many regressors, as well as an alternative way to model the time-dependency, here through state space models (see Section 2.4.6). In order to implement the Bayesian Structural Time Series model, the package `bsts`[9] developed by Google was used. For all BSTS models, 4500 samples were drawn in each step. This is in accordance with Scott et al. [51], in which the study used 5000 samples. To account for the burn-in, the first 1,000 samples were discarded, yielding a remaining 3,500 draws to perform inference on. This was in accordance with Gelman [38], who claims that at least 2,500 samples are required for an accuracy of about 1 %. We therefore added another 1,000 to this, ensuring a 1 % accuracy.

A disadvantage with the `bsts` package is that there are limitations on how to specify the priors. In this model, *Spike-and-slab* priors are used. These can be described as priors that contribute to which predictors are selected in the end. One can also suggest probable values, and set the strength of the prior in terms of number of observations that the prior is built on (`prior.information.weight`). For example, having 157 observations, and setting the prior to being based on 157 observation made the information from the data and the prior equally significant.

The package allowed setting prior inclusion probabilities through the argument `prior.inclusion.probabilities`, and these were set to 1 for the media variables and 0.5 to the rest. Furthermore, the argument `expected.model.size`, including prior information on the number of regressors to include, were set to 16. For the other priors, the standard settings were used. Although this is generally not recommended by D. McNeish [58], this was done as we were not completely sure of how to set these. We then set the strength of the priors (through `prior.information.weight`) to 0.01 observations, being the default of the package. The priors therefore did not have a strong influence in comparison to the data, and could then be considered weak priors.

The BSTS model experiments began with an `auto.arima` procedure from the package `forecast` in R[10]. It was done to specify an order of the possible AR-process to be included in the model. After this, an initial model on the whole data set is created using an equal number of MCMC samples as used in the coming steps. From this fitted model, inclusion probabilities are extracted and predictors are chosen. After this, a time series cross-validation

[9] Documentation available at `https://cran.r-project.org/web/packages/bsts/bsts.pdf`. Accessed 18/6 - 2019.

[10] Documentation available at `https://cran.r-project.org/web/packages/forecast/forecast.pdf`. Accessed 18/6 - 2019.

of the model was performed, which determined the confidence intervals of the parameters, ROIs of the media coefficients and the fit of the model.

For the BSTS models, an initial model including all regressors was fitted, with which a model with probabilities for incorporating each predictor is obtained. Based on these probabilities, predictors that must be included (e.g., media spendings and intercept) and the additional predictors with the highest probabilities were chosen for further modelling.

3.3.5.2 Bayesian hierarchical models

On the real-world data set where a finer geographical granularity is available, a Bayesian hierarchical model was implemented. The goal of this model was to explore the use of prior information as well as the possibility of lowering uncertainty by modelling on a store-basis rather than on a country-basis. The hierarchical model which was implemented can be stated as following, with the relation and priors for store j stated as

$$\begin{aligned} y_j &= X_j\beta_j + \varepsilon_j \\ \beta_j &\sim \mathcal{N}(\Delta, V_\beta) \\ \varepsilon_j &\sim \mathcal{N}(0, \tau_j) \end{aligned} \tag{3.7}$$

and hyper-priors stated as:

$$\begin{aligned} \tau_j &\sim \nu_j \text{ssq}_j / \chi^2_{\nu_j} \\ V_\beta &\sim IW(\nu, V) \\ \Delta &\sim \mathcal{N}(\bar{\Delta}, V_\beta \otimes A^{-1}) \end{aligned} \tag{3.8}$$

This model was chosen by assuming normal errors for the relation $y_j = X_j\beta_j + \varepsilon_j$ for each store and in turn choosing conjugate priors, cf. Rossi et. al. [66], for such a model. The implementation of this model was done through the R package `bayesm`[11], which has an implementation of a Gibbs sampler for the described model. The sampling was run with 22000 samples and the first 2000 samples were discarded as a burn-in period. This was done to account for the possibility that the starting point is in a non-probable region. Out of the remaining 20000 samples, every 5th was kept in order to reduce memory usage, ending up with a total of 4000 samples. From these samples, the predictions were done through the posterior mean of the coefficients. Further, the ROI calculation was restricted to the last 2000 of these samples, due to time constraints, since this had to be done for each store. As point estimate on a store basis, the posterior mean was used, as this was deemed suitable as according to the discussion in Section 2.4.1.1.

Due to varying sizes of sales-numbers for the stores, both the predictors and response variables were standardised by subtracting the mean and dividing by the empirical standard deviation. This was done in order to put sales-numbers on the same scale, which make the coefficients more comparable.

Priors play a large roll in Bayesian models and thus three different priors were tested: a vague set of priors, a generic weakly informative set of priors and an informative set of priors based on the specific problem. The expressions can be slightly reformulated. For $\nu > p + 1$ we have that $\mathbb{E}[V_\beta] = \frac{V}{\nu - p - 1}$, cf. Mardia et. al. [57], and as a result the parameter V can be reformulated as $V = \Sigma \cdot (\nu - p - 1)$ for some expected covariance matrix Σ. An interpretation of the parameter can be seen in Table 3.3.

The parameters $\bar{\Delta}$, Σ, ν and A were changed for the different priors, whereas ν_j and ssq_j were kept at their default values given by Rossi et. al. [66]: $\nu_j = 3$ and $\text{ssq}_j = Var(y_j)$.

The vague priors were set to have a small confidence in the parameters suggested by the prior. As a result, ν and A were small whereas σ_1^2 was large. This results in a model with a prior that does not have much influence. The generic weakly informative case can be seen as applying a regularization. Like in the vague case, $\bar{\Delta}$ was set to $\mathbf{0}$ in order to push parameter

[11] `https://cran.r-project.org/web/packages/bayesm/bayesm.pdf`. Accessed 18/6 - 2019.

Table 3.3: Interpretation of hyper-prior parameters

Parameter	Interpretation
$\bar{\Delta}$	Expected value of Δ, which in turn is the expected value of β_j
Σ	The Expected value of the covariance matrix V_β.
ν	Effectively represents the certainty in the covariance. Larger value suggest higher confidence.
A	Represents the confidence in the accuracy of $\bar{\Delta}$. Larger value suggest higher confidence.

Table 3.4: Priors used in experiments of the Bayesian hierarchical model. p is the number of predictors.

	$\bar{\Delta}$	Σ	ν	A
Prior 1	$\mathbf{0}$	$\mathbb{I}$	$p+3$	0.01
Prior 2	$\mathbf{0}$	$\mathbb{I} \cdot \frac{1}{1000}$	$p+5000$	5000
Prior 3	$\bar{\beta_j}$	$Cov(\beta_j)$	$p+5000$	5000

estimates toward zero, however, the confidence in the parameters were set a lot higher. ν and A were set to higher values compared to the vague case and σ_2^2 were smaller than σ_1^2.

Lastly the informative prior took a different route. The data set consists of store-level data from three countries: Sweden, Denmark and Norway. Since the modelling is done on the Swedish data, the Norwegian and Danish data were used to form an informative prior, by assuming the relations are similar between these countries. A least squares fit was applied to each store (+60 stores) of these two countries. $\bar{\Delta}$ was then set to be the sample mean of the estimated β_j and Σ was set to be the sample covariance. A summary of the parameters can be seen in Table 3.4. Note that the strength of the priors have not been studied and conclusions are only based on these specific priors.

3.3.6 XGBoost

In order to explore the use of a non-parametric model in an MMM context, XGBoost was implemented. The investigation of XGBoost differed from the investigation of the parametric models, mainly due to XGBoost's nature being a non-parametric model. Multicollinearity becomes slightly irrelevant, as this is rather a problem for parametric models where one attempts to directly relate contributions of predictors through parameters.

The data was not transformed to suit a different functional form than the regular one; in a marketing context, the data transformation is made due to the advantage of providing a saturation effect for parametric models. Since XGBoost is not a parametric model, a data transformation can be regarded unnecessary.

3.3.6.1 Tuning parameters

There are a few hyper parameters that were to be set. First of all, the number of trees were set to less than half the number of data points, as well as the maximum number of leaves in each tree. The number of leaves in each tree L, as well as the depth were also limited. While the depth was limited to half of the number of predictors, the number of leaves was limited to half of the number of data points, again to try and avoid overfitting. Thus, an XGBoost model was

fitted and has its parameters tuned using the function `RandomizedSearchCV`[12] available in the package `sklearn`. Random searches has previously been proven to be superior to exhaustive grid searches for parameter tuning while finding its optimal solution within a fraction of the computational time [4]. To select model, the one receiving the lowest Cross-Validated Mean-Squared Error-value was chosen out of 10,000 iterations. The possible parameters used in this randomised search can be seen in Table 3.5. Explanations of these can be found in the documentation of the package[13].

Table 3.5: Set of XGBoost parameters used in the randomised search

Parameter	Values or distribution to sample from
`colsample_bytree`	$\{0.2, 0.4, 0.5, 0.7, 1\}$
`gamma`	$Unif(0, 1)$
`learning_rate`	$Reciprocal(10^{-6}, 1)$
`max_depth`	$\{1, 2, ..., 8\}$
`n_estimators`	$\{4, 5, .., 100\}$
`reg_alpha`	$Unif(0, 10)$
`reg_lambda`	$Reciprocal(10^{-5}, 10)$
`subsample`	$Unif(0, 1)$

In Table 3.5, $Unif(a, b)$ refers to the uniform distribution between a and b. $Reciprocal(a, b)$ refers to the reciprocal distribution, with a probability density function defined as seen in 3.9. This is a convenient distribution to sample values of different orders of magnitude uniformly, something convenient for hyperparameters.

$$p(x|a, b) = \frac{1}{x\big(\log(b) - \log(a)\big)} \tag{3.9}$$

Below follows an explanation of all hyper parameters[14].

- `colsample_bytree` refers to the fraction of columns sampled for each tree, with which each tree is constructed with. `gamma` gives the smallest loss reduction that is needed to make another partition when constructing a tree. A higher value of `gamma` will lead to less partitioning.
- `learning_rate` defines the step size in the learning of the boosting. This shrinks the weights of the features to prevent overfitting and thus performs a form of regularisation.
- `max_depth` is the maximum depth of each tree in the algorithm.
- `n_estimators` refers to the number of trees combined in the algorithm.
- `reg_alpha` and `reg_lambda` are respectively L1- and L2-regularising terms on the weights. The higher the value, the more conservative the algorithm will be.
- `subsample` sets the number of training points to sample out of all data points while constructing the trees.

3.4 Evaluation

While quality of fit measurements are usually easily comparable between the models, measures regarding uncertainty and interpretability can be less so, as methods of deriving these can differ vastly.

[12] See `https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.RandomizedSearchCV.html`. Accessed 18/6 - 2019.

[13] Available at `https://xgboost.readthedocs.io/en/latest/parameter.html`. Last retrieved on 18/6 - 2019

[14] Explanations available at `https://xgboost.readthedocs.io/en/latest/parameter.html`. Last retrieved on 18/6 - 2019

3.4.1 Prediction performance

In order to measure the prediction performance of the models a few different metrics were used. These metrics are the Mean Absolute Error (MAE, Equation 3.10) Root Mean-Squared Error (RMSE, Equation 3.12), the Mean Average Percentage Error (MAPE, Equation 3.11) and the R^2-value (Equation 3.13). The MAE and RMSE were chosen as they can be used in conjunction to evaluate the models to provide more information. For example, a small RMSE and a large MAE can indicate that the model overfits to a small number of data-points. Comparison between the MAE and the RMSE can provide further insights on the variation of the error. The MAPE, was chosen as it is especially useful for comparisons between data sets due to its scaling-invariance, as it provides the average percentage of deviation from the true values rather than an absolute number.

While the R^2-value provides a measure on how much of the variance is explained by the model, and does not necessarily tell whether the model is biased or not, the RMSE measures the risk of the model through determining the average deviation from the actual values. A combination of these values results in a discussion to determine which model is fitted most appropriately.

One must note that the R^2-value was only used as a measure on the linear and intrinsically linear models. This is due to the fact that the R^2-value can be considered inadequate for measuring the quality of fit in a non-linear setting, as proven by Spiess et. al [68]. Thus, the quality of fit of XGBoost and the hierarchical models were determined by the MAE, MAPE and the RMSE.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{3.10}$$

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \tag{3.11}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y})^2} \tag{3.12}$$

$$R^2 = \frac{\sum_{i=}^{n}(\hat{y}_i - y_i)^2}{\sum_{i=1}^{n}(y_i - \overline{y})^2} \tag{3.13}$$

Note that the RMSE, MAE and MAPE were evaluated on the actual response predictor's value and not in the transformed predictor's measure, in order to make the different models comparable. The metrics are obtained by the time-series cross validation (See Algorithm 3). This can also be seen as using the lead-1 forecast of each point in the test set to receive the metrics. The initial test size was set to be the last year, out of the three total. This is to ensure a large enough training set for the parameters, but also to make sure there are at least two observations for each seasonal period. The metrics evaluated on the training data was evaluated on the original training set (the first two years). However, the R^2 was retrieved from the fit of each model that is fitted on the whole data set.

3.4.2 Uncertainty evaluation

The evaluation of mitigation of uncertainty is of course highly related to the validation the results as a high uncertainty reduces confidence in the model. Evaluating the uncertainty, and mitigation thereof, is a hard task which differs strongly from model to model. As a result, different methods were applied to the different categories of models and the evaluation slightly differed.

3.4.2.1 Linear models

For the (intrinsically) linear models, the uncertainty in parameter estimates were measured through the use of bootstrap confidence intervals. The blockwise bootstrap was implemented and used as by Fitzenberger [36] and Romano and Wolf [65], in a nonparametric way by resampling blocks $\{(X_t, y_t), ..., (X_{t+l}, y_{t+l})\}$ of the predictors and response variables. Bühlmann [15] mentions that $l \approx 2n^{(1/3)}$ seems to be good choice for block length in multiple cases. As a result the block length was chosen to $l = 10 \approx 2 \cdot 156^{(1/3)}$. As the method is non-parametric, this allowed for not having to make any assumptions of the distributions on the parameters. As the ROI-estimates in turn are derived from the parameter estimates, the non-parametric

The bootstrap methods were used to construct confidence intervals of parameter estimates. These intervals were constructed by the percentile based bias-corrected intervals, see Section 2.10.4. Efron and Hastie [35] mention that these intervals require bootstrap samples in the order of 2000. As a result, the amount of bootstrap samples were 2000 for all experiments.

3.4.2.2 Bayesian models

In the Bayesian setting, the confidence intervals were retrieved from the sampled posterior distribution received through MCMC sampling of different sorts. Estimation of parameters and therefore predictions were made using the posterior mean, being the most suitable as discussed in Section 2.4.1.1. In the BSTS models, the 1,000 first samples were discarded, which yielded a remaining 3500 samples. In the Bayesian hierarchical model, a burn-in of 2,000 was made as discussed previously, and only every fifth draw were used for inference, yielding 4,000 draws to perform inference on.

3.4.2.3 XGBoost

As discussed in Section 2.5, coefficient estimates are not obtainable as parameters for each predictor do not exist. Instead, other method have to be used in order to measure uncertainty in the model. This was done through extracting the ROI:s through the use of SHAP values with the method explained in the section below. These were evaluated through a standard, non-parametric bootstrap (see Algorithm 4) with 2,000 samples.

3.4.3 Evaluating certainty of ROI estimates

The 95 % confidence intervals for each model's ROI estimates for the media predictors were studied to see which model is the most confident in estimating this measure. However, the reasonableness of the value was also be taken into account. For example, if a model displayed negative ROIs for all media predictors, the model was considered inadequate as marketing can be assumed to have a positive effect on sales.

In terms of calculating the ROI, the methods presented in Section 2.8.3 were used for all models, except for XGBoost, as it is a non-parametric regression model. For the linear functional form, the linear method was used. In other words, the ROI corresponded to the β-coefficient for each media predictor. For the log-linear and the semi-logarithmic functional forms, the method proposed by D. Garg et al. [27] was used. The methods did however differ slightly between models; for GLS, the contributions of the ARMA-processes were also taken into account by including the contributions to the predictions of the AR- and MA-processes. However, the methods were in essence the same, just sometimes including more parameters.

For XGBoost, the SHAP values were used as an attempt to measure the aggregated ROI. Due to the SHAP values' theoretical nature, these should theoretically provide adequate measures on how much a predictor (e.g., a media spending) contributed to the response predictor at each time-point as presented in section 2.6. Thus, a linear fit's β-coefficient of the media spend to the media spend's SHAP values should yield an approximation of the ROI. Thus, assume that $SHAP_{m,t}$ is the SHAP-value for media channel m during time t, and $x_{m,t}$ be

the spend in media channel m during time t. The ROI for media channel m can then be approximated as

$$ROI_m \approx \beta_m, \quad \beta_m \in SHAP_{m,t} = \beta_0 + \beta_m x_{m,t} \tag{3.14}$$

This method therefore provided a linearization of the calculation of the ROI, just like the method presented by D. Garg et al. [27] as well as the linear functional form. Although another method to calculate the ROI:s exists, as presented in 2.8.3, the methods here were chosen due to their simplicity and after discussions with employees at Nepa.

4 Results

4.1 Model-specific parameters

The found order of the $ARMA(p, q)$-processes for both the simulated and the real-world datasets can be seen in Table 4.1.

Simulated data		**Real-world data**	
Model	Order (p,q)	Model	Order (p,q)
Linear	(0,0)	Linear	(0,0)
Log-linear	(0,3)	Log-linear	(0,1)
Semi-logarithmic	(2,2)	Semi-logarithmic	(0,1)

Table 4.1: The ARMA orders identified during time-series analysis.

The parameters found for XGBoost through the randomised search can be seen in Table 4.2.

Simulated data		**Real-world data**	
Parameter	Value	Parameter	Value
`colsample_bytree`	0.964	`colsample_bytree`	1
`gamma`	0.9172	`gamma`	0.738
`learning_rate`	0.1685	`learning_rate`	0.558
`max_depth`	6	`max_depth`	7
`n_estimators`	89	`n_estimators`	88
`reg_alpha`	0.271	`reg_alpha`	7.21
`reg_lambda`	$8.1 \cdot 10^{-3}$	`reg_lambda`	$7.14 \cdot 10^{-3}$
`subsample`	0.8026	`subsample`	0.939

Table 4.2: The parameters identified for XGBoost through random search cross validation.

4.2 Simulated data

Below are the results for the simulated dataset displayed. First, the results for the quality of fit is presented, followed by measurements of uncertainty. The uncertainty measurements are

displayed as bias-corrected confidence intervals where the utmost points represent the 95 % confidence interval, while the box represents 25 and 75 % quantiles.

Note that only the bootstrap confidence intervals have been bias-corrected through the method described in Section 3.4.2, and not the confidence intervals retrieved through MCMC sampling.

4.2.1 Quality of fit

The prediction performance, according to the measures introduced in Section 3.4.1, can be seen in Table 4.3. The RMSE of the different models can be seen in Figure 4.2,

Model	RMSE	MAE	MAPE	R^2
Linear OLS	$3.301 \cdot 10^5$	$2.457 \cdot 10^5$	8.45%	0.788
Log-linear OLS	$3.612 \cdot 10^5$	$2.685 \cdot 10^5$	9.12%	0.805
Semi-logarithmic OLS	$3.541 \cdot 10^5$	$2.52 \cdot 10^5$	8.7%	0.7905
Log-linear GLS	$3.175 \cdot 10^5$	$2.559 \cdot 10^5$	8.91%	0.912
Semi-logarithmic GLS	$3.249 \cdot 10^5$	$2.629 \cdot 10^5$	9.25%	0.8996
Linear robust	$3.306 \cdot 10^5$	$2.445 \cdot 10^5$	8.47%	0.787
Log-linear robust	$3.566 \cdot 10^5$	$2.645 \cdot 10^5$	9.09%	0.802
Semi-logarithmic robust	$3.502 \cdot 10^5$	$2.485 \cdot 10^5$	8.66%	0.789
Linear constrained	$4.911 \cdot 10^5$	$3.74 \cdot 10^5$	14.74%	0.484
Log-linear constrained	$4.036 \cdot 10^5$	$3.138 \cdot 10^5$	12.52%	0.758
Semi-logarithmic constrained	$3.113 \cdot 10^5$	$2.555 \cdot 10^5$	8.93%	0.899
Linear Shapley	$3.425 \cdot 10^5$	$2.675 \cdot 10^5$	9.37%	0.742
Log-linear Shapley	$3.483 \cdot 10^5$	$2.699 \cdot 10^5$	8.95%	0.765
Semi-logarithmic Shapley	$3.486 \cdot 10^5$	$2.748 \cdot 10^5$	9.4%	0.7454
Linear BSTS	$2.877 \cdot 10^5$	$\mathbf{2.226 \cdot 10^5}$	7.78%	**0.9681**
Log-linear BSTS	$3.413 \cdot 10^5$	$2.728 \cdot 10^5$	9.36%	0.798
Semi-logarithmic BSTS	$3.26 \cdot 10^5$	$2.555 \cdot 10^5$	8.83%	0.805
XGBoost	$\mathbf{2.799 \cdot 10^5}$	$2.232 \cdot 10^5$	**7.54**%	N/A

Table 4.3: Result of prediction measures on the test set (R^2 calculated on whole set). Best results are marked in bold. Note that R^2 for XGBoost was intentionally not calculated.

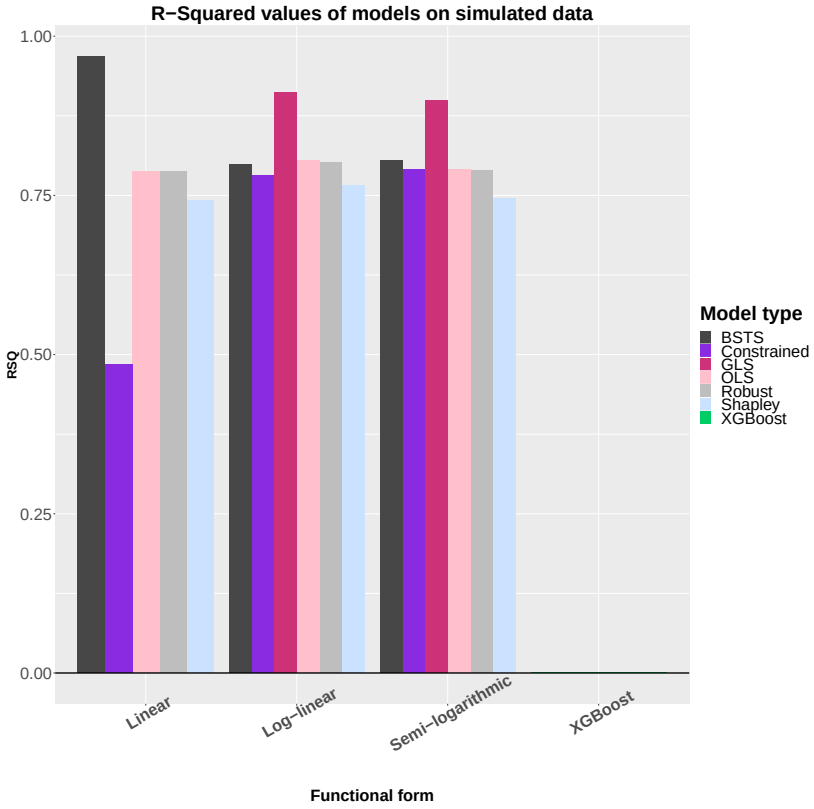


Figure 4.1: The R^2-value of the models applied on the simulated data.

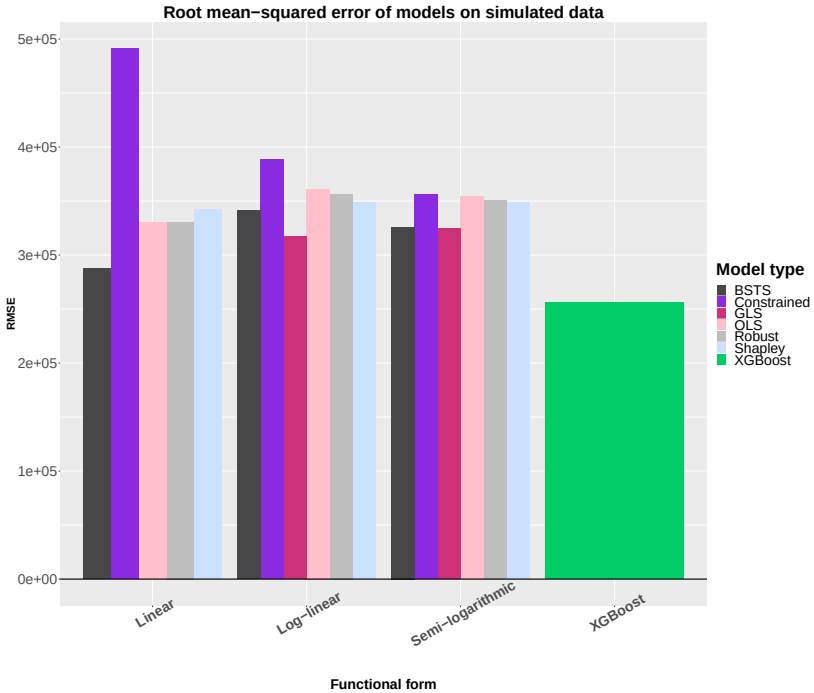


Figure 4.2: The Root Mean-Squared error of the different models on the simulated test data. Note that hierarchical models were not implemented here since no hierarchical structure was made for the simulated data.

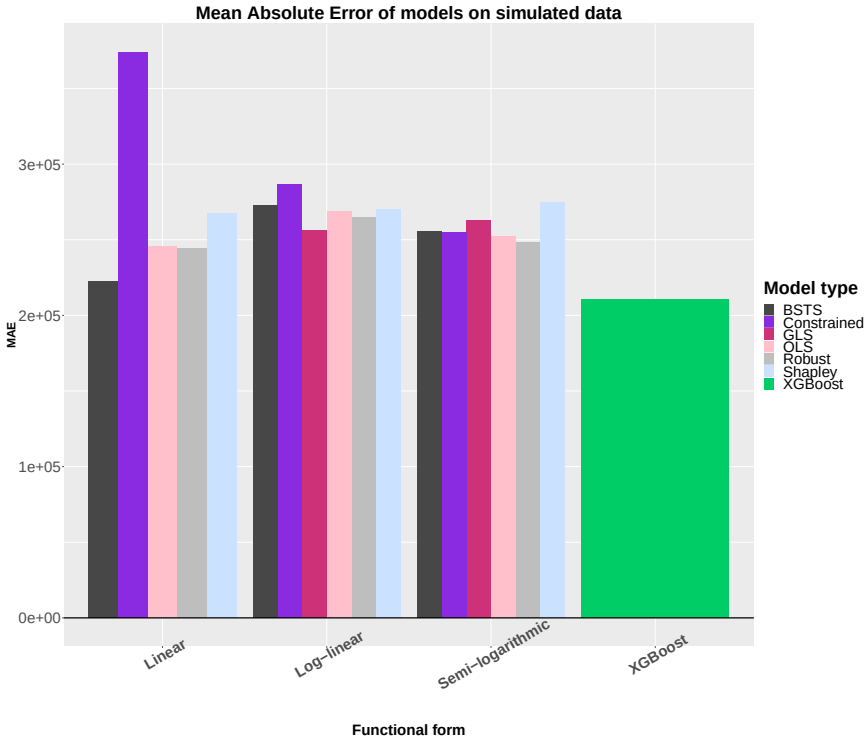


Figure 4.3: The Root Mean Absolute error on the simulated test data.

4.2.2 Uncertainty measurements

The confidence intervals for the media spend coefficients are shown first in this section. This is followed by the 95 % confidence intervals of the ROI estimates according to the method stated in section 3.4.3. Note that different methods are used in order to estimate the confidence intervals for different modelling methods (see legends in each plot). Mainly, the posterior is used for the Bayesian methods whereas the blockwise bootstrap is used for the non-Bayesian methods. All uncertainty measurements are displayed as confidence intervals where the utmost points represent the 95 % confidence interval, while the box represents 25 and 75 % quantiles.

Note that only the bootstrap confidence intervals have been bias-corrected through the method described in section 3.4.2, and not the confidence intervals retrieved through MCMC sampling, as this is not needed.

4.2.2.1 Coefficient estimates

Below, the coefficient estimates for the media variables `display...net`, `facebook...net` and `search_branded` are shown. Note that no linear ARMA-process was found for the linear process, hence the absence of GLS in the linear form. The constrained model were removed in Figures 4.4 and 4.5 due to having too wide confidence intervals.

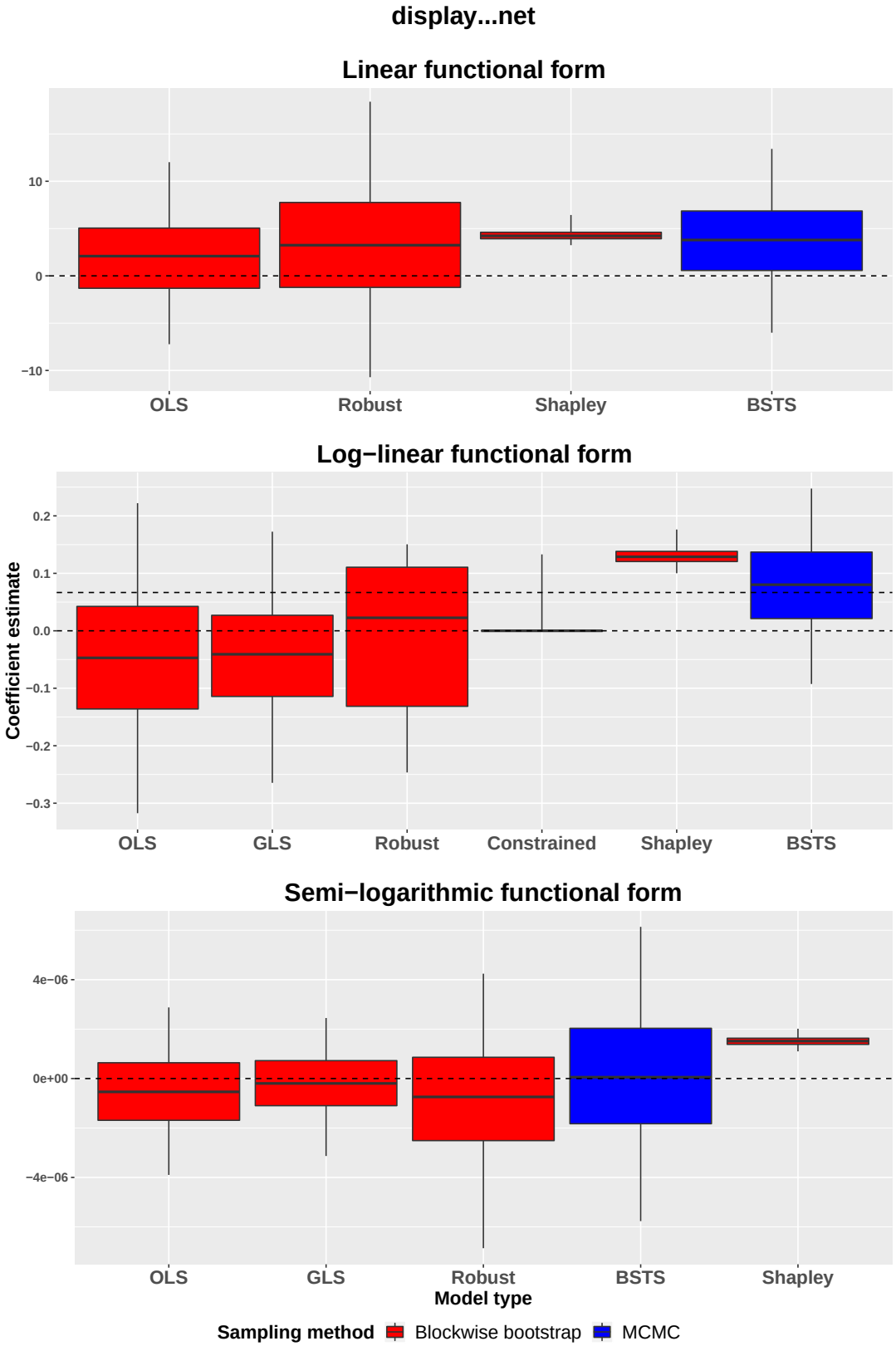


Figure 4.4: The confidence intervals for the coefficient estimates for the Display predictor. The dashed lines shows where 0 is. In the log-linear form, the second dashed line represents the true value. A tighter box indicates a higher confidence.

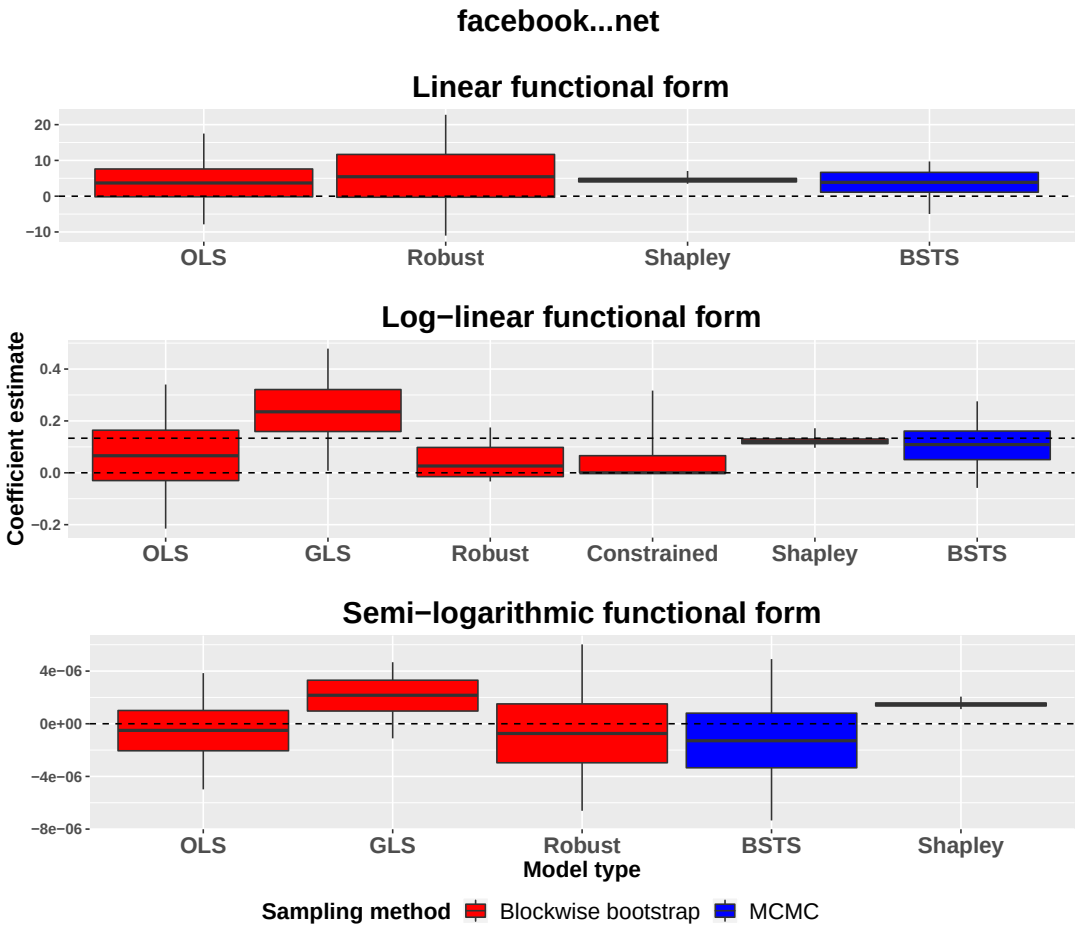


Figure 4.5: The confidence intervals for the coefficient estimates for the Facebook predictor. The dashed lines shows where 0 is. In the log-linear form, the second dashed line represents the true value. A tighter box indicates a higher confidence.

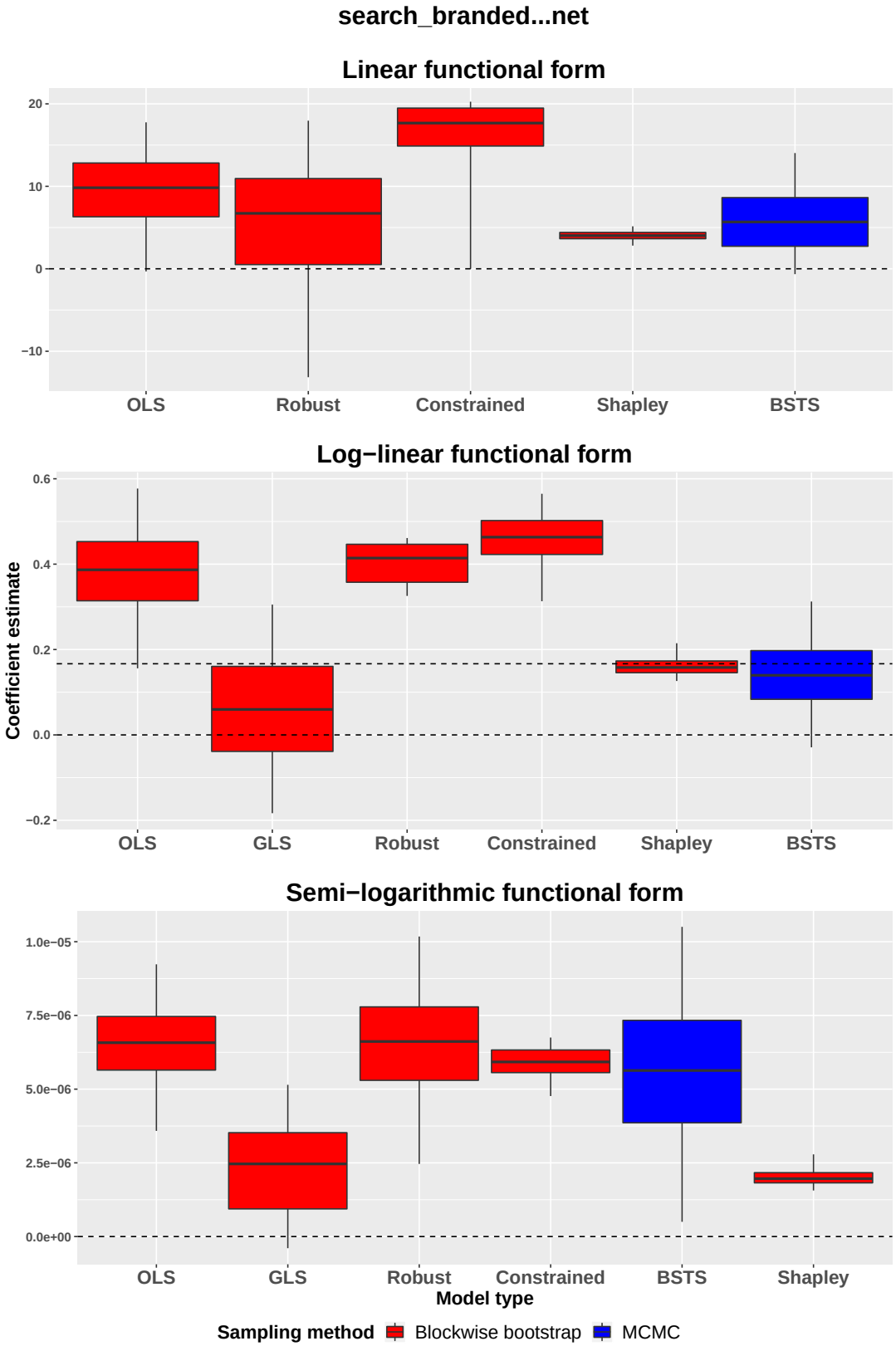


Figure 4.6: The confidence intervals for the coefficient estimates for the Search Branded predictor. The dashed lines shows where 0 is. In the log-linear form, the second dashed line represents the true value. A tighter box indicates a higher confidence.

4.2.2.2 Return on investment estimates

Figure 4.7 show the ROI estimates for the variables `display...net`, `search_branded...net` and `facebook...net`.

For all predictors, the ROI estimates were removed for the log-linear OLS, log-linear GLS, log-linear robust, log-linear constrained, log-linear Shapley, the linear constrained and the log-linear BSTS models. This since these showed too wide confidence intervals and were deemed unreasonable.

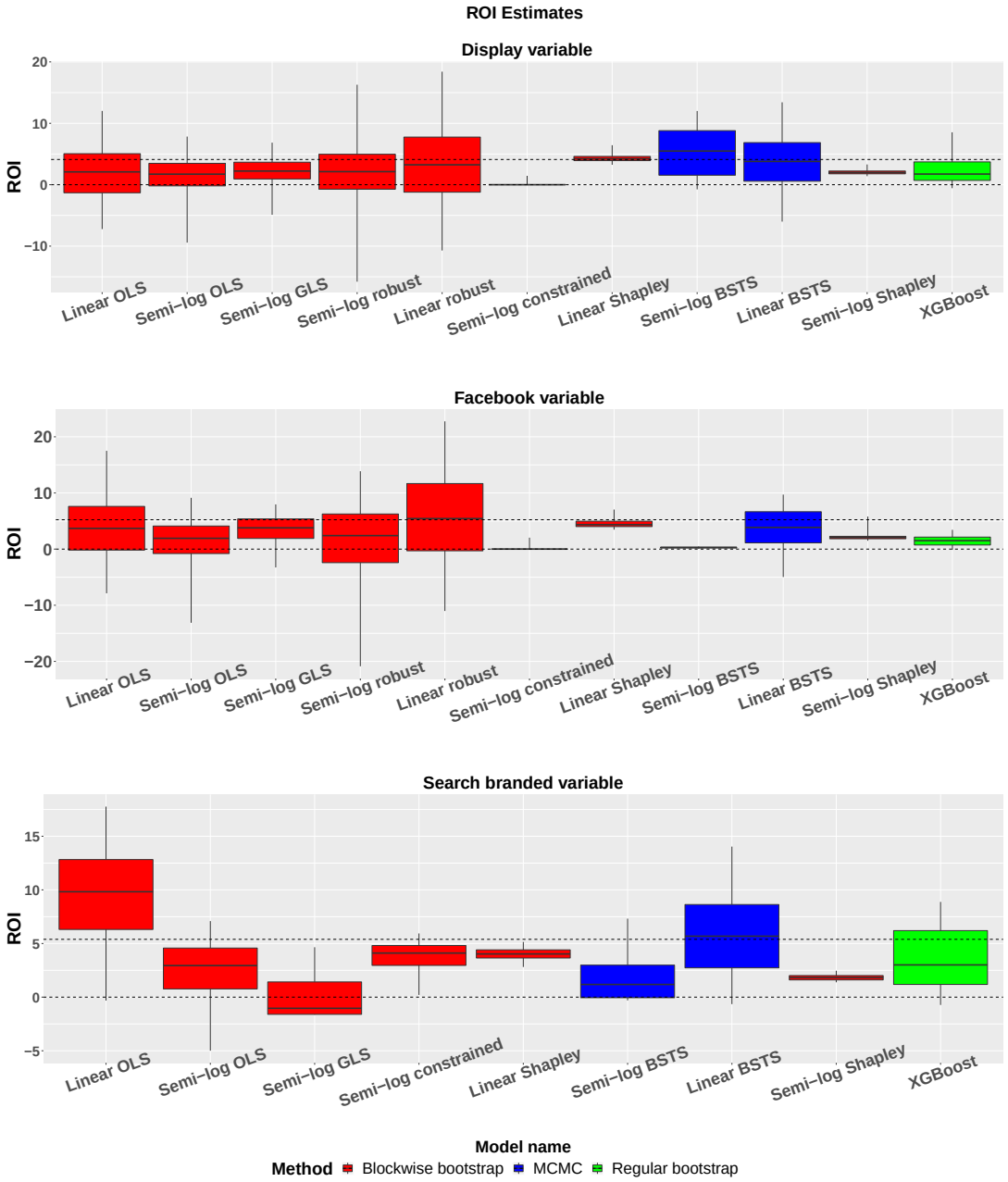


Figure 4.7: The confidence intervals for the ROI-estimates for the media predictors. Note that all models' estimates are not included due to some having too wide intervals. One dashed line shows where 0 is located - the other displays the true ROI value.

4.3 Real data

Below, the results for the real dataset are displayed. First, the results for the quality of fit are presented, followed by measurements of uncertainty.

4.3.1 Quality of fit

Table 4.4 displays the quality of fit measurements used for evaluating the models. Furthermore, Figure 4.8 shows the R^2-values for the models for which the R^2-value can be considered an adequate measure. For the other models, this value was not calculated, as it can be deemed misleading for non-linear models, as discussed by Spiess et al. [68].

Model	RMSE	MAE	MAPE	R^2
Linear OLS	$4.184 \cdot 10^7$	$2.674 \cdot 10^7$	26.12%	0.901
Log-linear OLS	$3.359 \cdot 10^7$	$1.857 \cdot 10^7$	15.03%	0.862
Semi-logarithmic OLS	$3.434 \cdot 10^7$	$1.893 \cdot 10^7$	14.81%	0.88
Log-linear GLS	$3.477 \cdot 10^7$	$1.904 \cdot 10^7$	15.22%	0.865
Semi-logarithmic GLS	$3.572 \cdot 10^7$	$1.911 \cdot 10^7$	14.72%	0.884
Linear robust	$4.429 \cdot 10^7$	$2.244 \cdot 10^7$	20.08%	0.857
Log-linear robust	$3.577 \cdot 10^7$	$1.834 \cdot 10^7$	14.66%	0.852
Semi-logarithmic robust	$4.088 \cdot 10^7$	$1.907 \cdot 10^7$	14.37%	0.86
Linear constrained	$4.16 \cdot 10^7$	$2.622 \cdot 10^7$	26.59%	0.908
Log-linear constrained	$3.51 \cdot 10^7$	$1.856 \cdot 10^7$	14.56%	0.851
Semi-logarithmic constrained	$3.417 \cdot 10^7$	$1.883 \cdot 10^7$	14.73%	0.88
Linear Shapley	$4.5 \cdot 10^7$	$2.283 \cdot 10^7$	21.89%	0.869
Log-linear shapley	$4.16 \cdot 10^7$	$1.818 \cdot 10^7$	14.23%	0.83
Semi-logarithmic Shapley	$4.133 \cdot 10^7$	$1.806 \cdot 10^7$	13.75%	0.857
Linear BSTS	$\mathbf{2.568 \cdot 10^7}$	$\mathbf{1.414 \cdot 10^7}$	13.23%	**0.974**
Semi-logarithmic BSTS	$4.542 \cdot 10^7$	$1.87 \cdot 10^7$	14.92%	0.6
Log-linear BSTS	$6.251 \cdot 10^7$	$2.679 \cdot 10^7$	19.88%	0.855
Linear hierarchical - vague prior	$3.907 \cdot 10^7$	$2.164 \cdot 10^7$	16.63%	N/A
Log-linear hierarchical - vague prior	$3.25 \cdot 10^7$	$1.635 \cdot 10^7$	11.85%	N/A
Semi-logarithmic hierarchical - vague prior	$3.398 \cdot 10^7$	$1.656 \cdot 10^7$	11.73%	N/A
Linear hierarchical - regularizing prior	$4.233 \cdot 10^7$	$2.11 \cdot 10^7$	15.63%	N/A
Log-linear hierarchical - regularizing prior	$4.267 \cdot 10^7$	$1.797 \cdot 10^7$	11.92%	N/A
Semi-logarithmic hierarchical - regularizing prior	$4.17 \cdot 10^7$	$1.67 \cdot 10^7$	**10.99**%	N/A
Linear hierarchical - informative prior	$3.841 \cdot 10^7$	$2.165 \cdot 10^7$	16.94%	N/A
Log-linear hierarchical - informative prior	$3.318 \cdot 10^7$	$1.684 \cdot 10^7$	12.26%	N/A
Semi-logarithmic hierarchical - informative prior	$3.381 \cdot 10^7$	$1.637 \cdot 10^7$	11.76%	N/A
XGBoost	$3.746 \cdot 10^7$	$2.025 \cdot 10^7$	17.7%	N/A

Table 4.4: Result of prediction measures on the test set (R^2 calculated on whole set). Best results are marked in bold. Note that R^2 for XGBoost and the hierarchical models were intentionally not calculated.

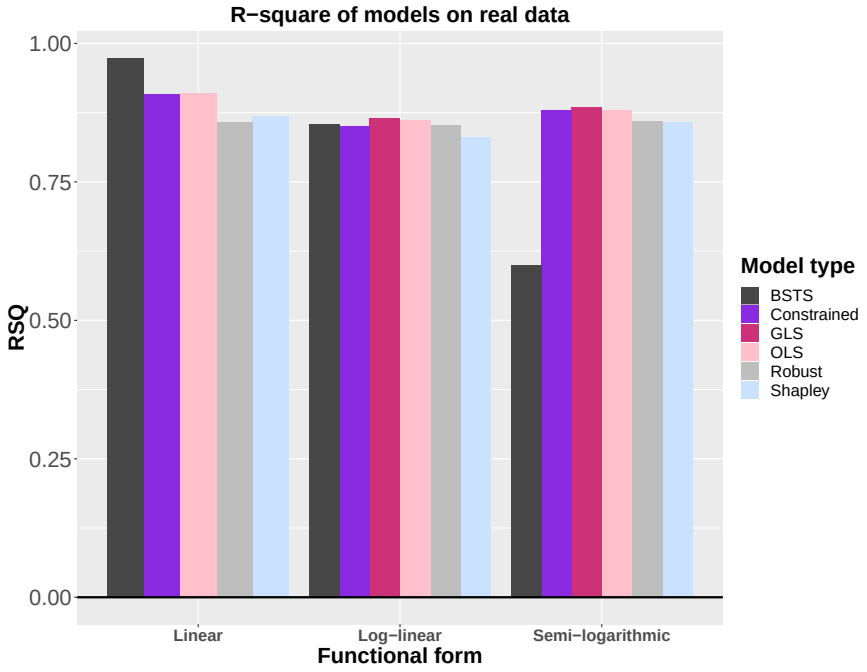


Figure 4.8: The R^2-value of the models applied on the real-world data. A higher value is better.

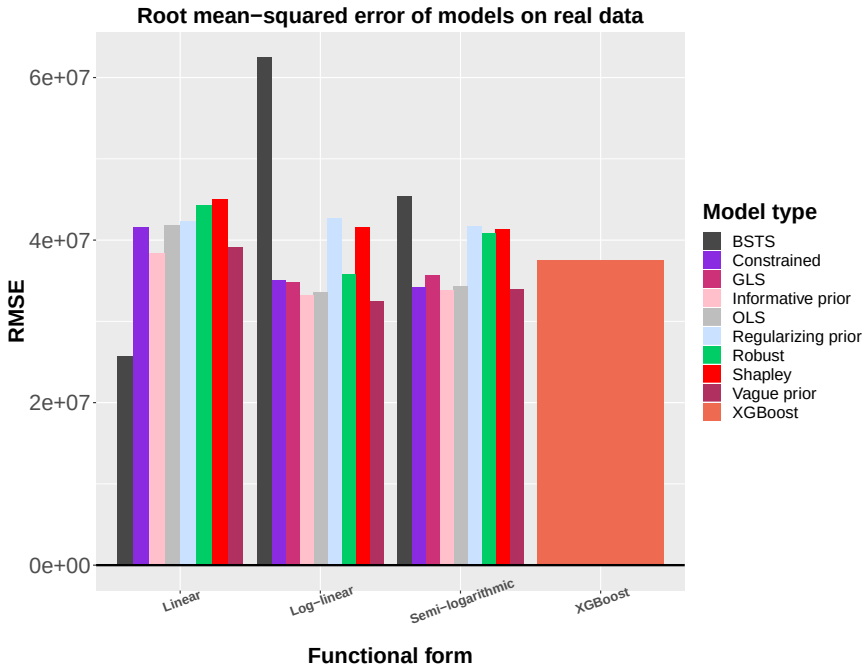


Figure 4.9: The Root Mean-squared error on the real-world test data. A lower value is better.

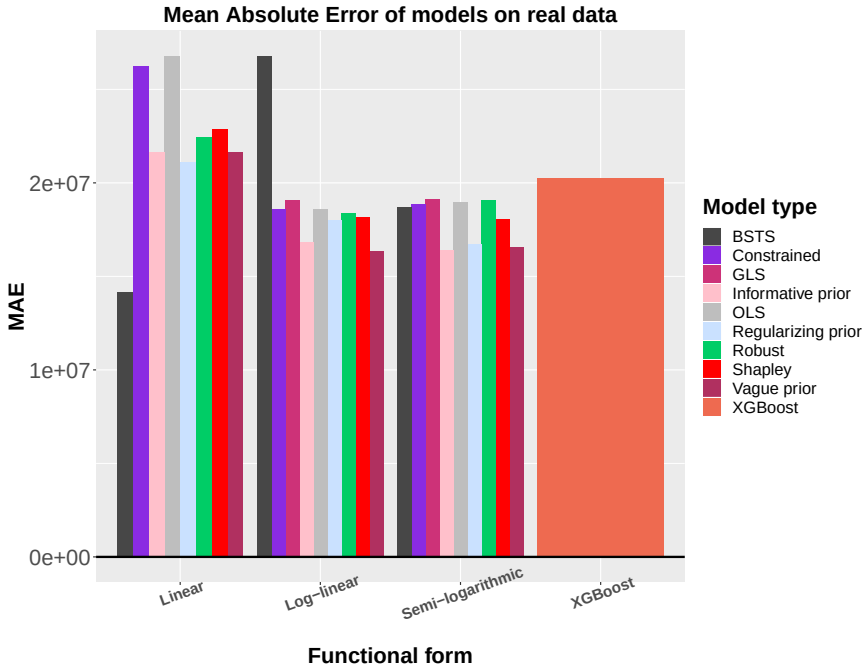


Figure 4.10: The Mean Absolute error on the real-world test data. A lower value is better.

4.3.2 Uncertainty measurements

4.3.2.1 Coefficient estimates

Below follows the coefficient estimates of the real data for the variables `TVC` (Figure 4.11) and `Facebook` (Figure 4.12). To avoid taking up too much space, not all confidence intervals on all media predictors are displayed. Instead, we have chosen a two predictors' confidence intervals to be representative for the results by either representing the typical scenario, or contradicting it. More results can be found in Appendix C. Also note that coefficient estimates for the hierarchical models are not displayed, as these include more than 40 parameter estimations for each variable (one for each store).

As can be seen in general, the OLS, GLS and Robust models generally produces similar confidence intervals, while the Shapley Value regression model consistently provides the tightest intervals.

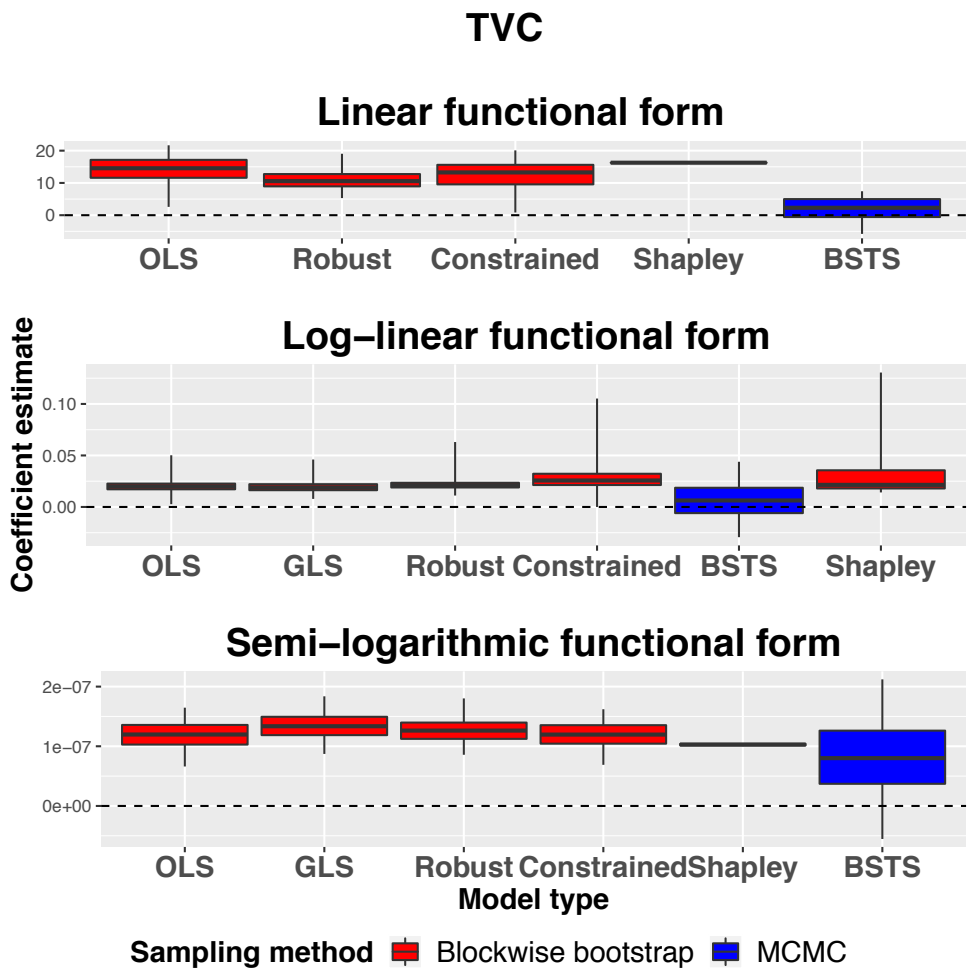


Figure 4.11: Coefficient estimates for the TVC predictor on the real data for the log-linear functional form. The dashed line shows where 0 is located.

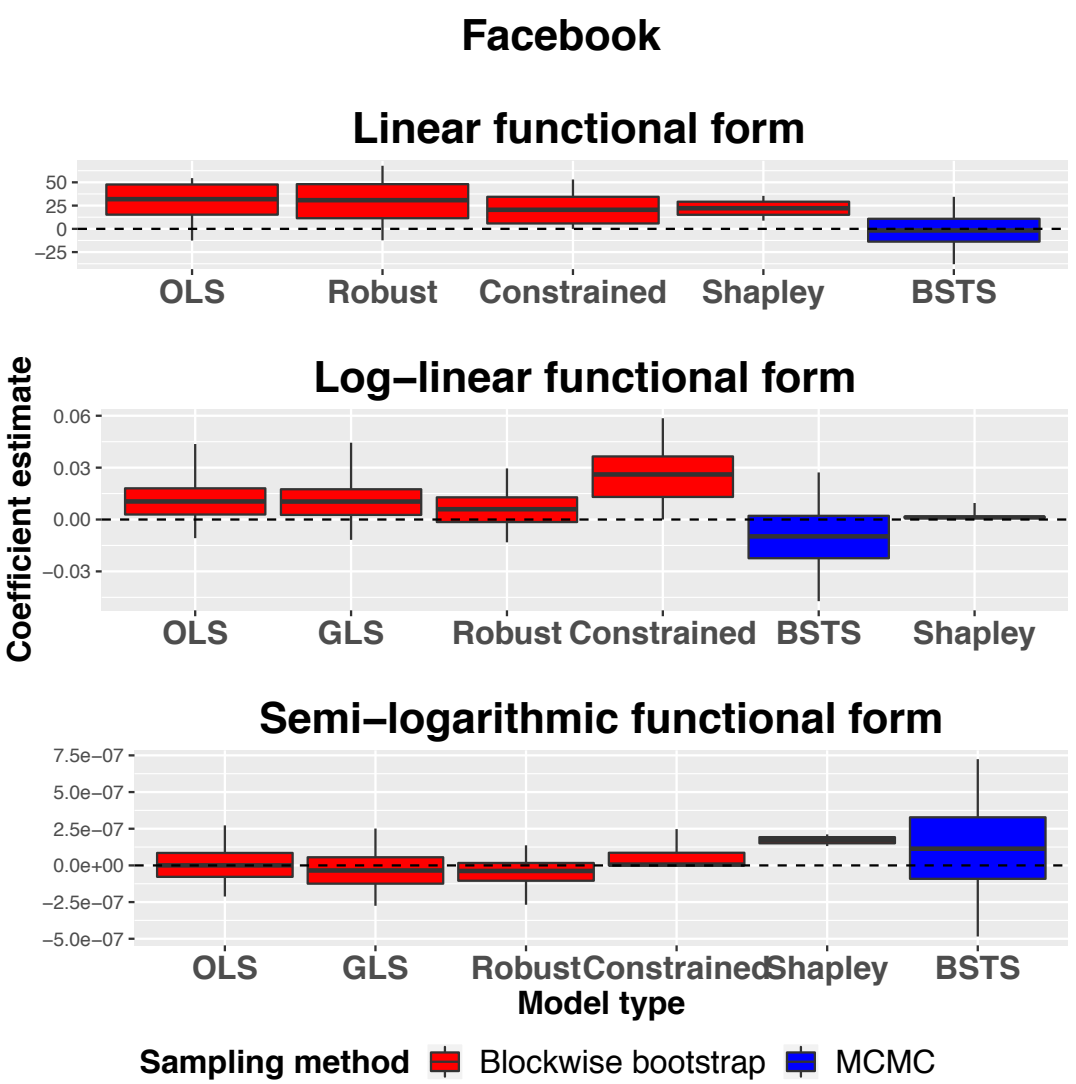


Figure 4.12: Coefficient estimates for the Facebook predictor on the real data. The dashed line shows where 0 is located.

4.3.2.2 ROI estimates

Below, the ROI estimates for the `DM` variable (Figure 4.13) and `Facebook` variable (Figure 4.14) are shown. Only two variables were chosen to represent what was commonly observed, followed by an example that contradicts this example. In general, the results were similar to the `DM` variable, although the `Facebook` contradicts the behaviour of the models seen in in Figure 4.13. The ROI estimates for XGBoost are here shown separately in Figure 4.15. The confidence intervals are very wide, always including the 0, indicating a high uncertainty in the estimation.

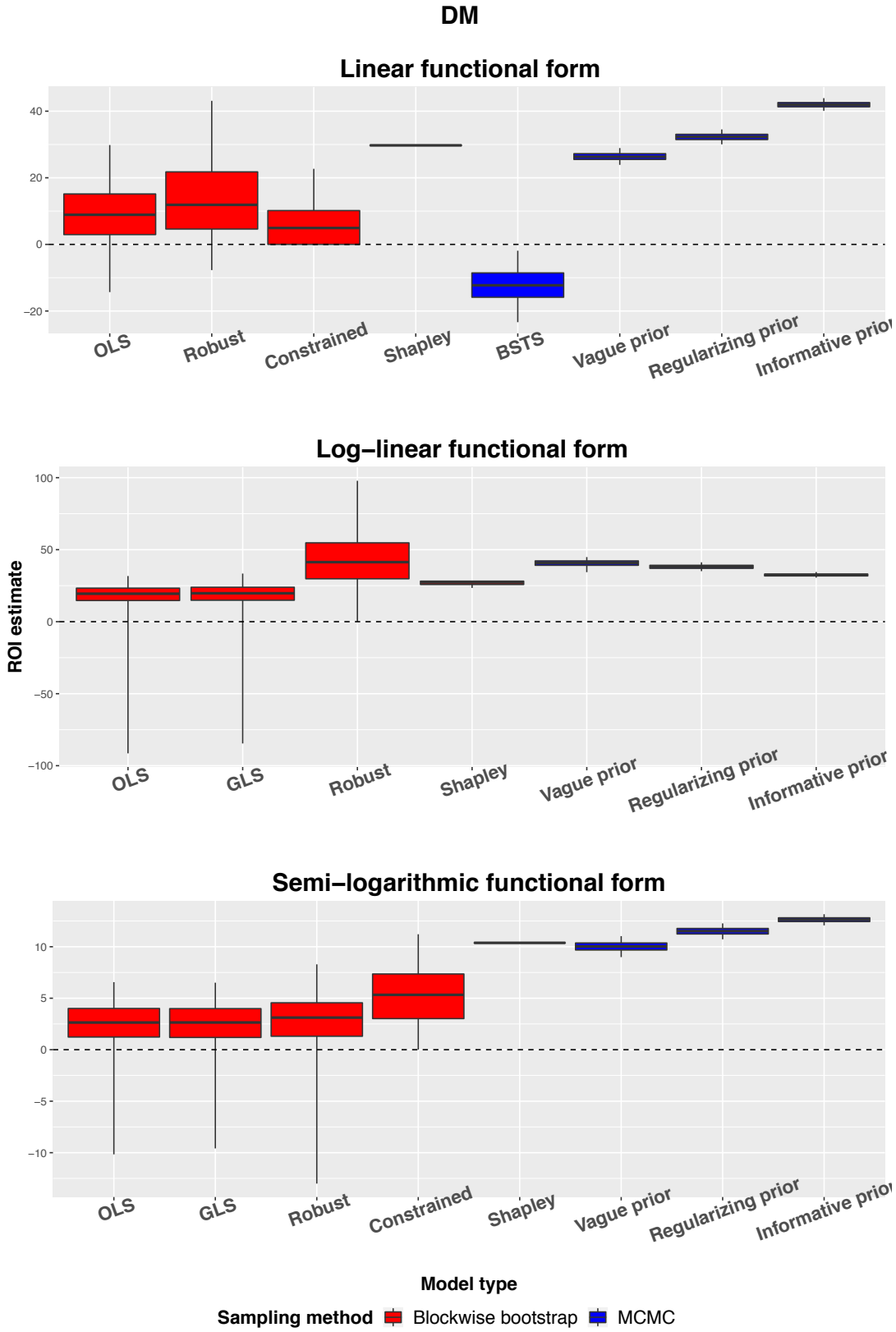


Figure 4.13: ROI estimates for the DM spend on the real data. The dashed line shows where 0 is located.

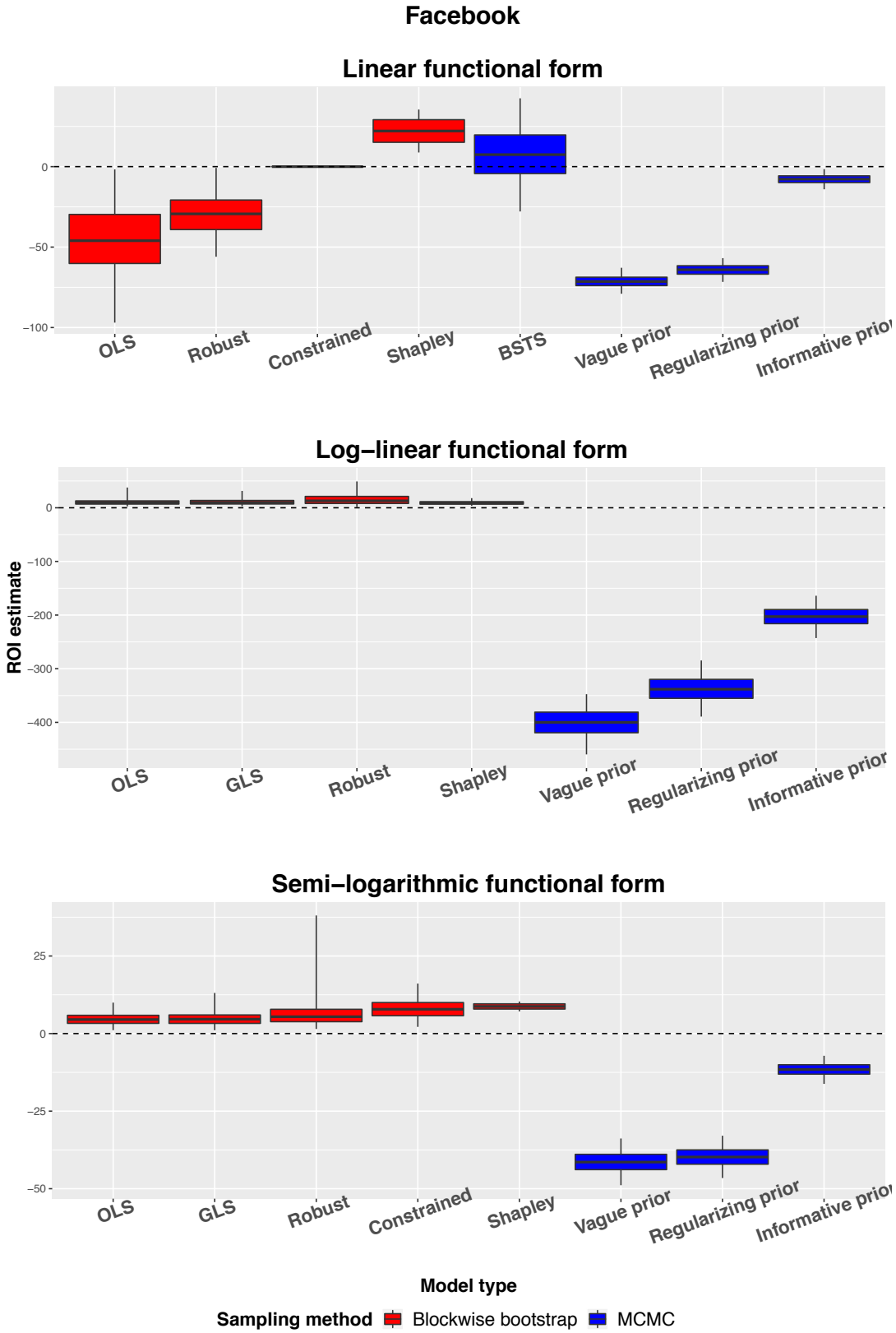


Figure 4.14: ROI estimates for the Facebook spend on the real data. The dashed line shows where 0 is located.

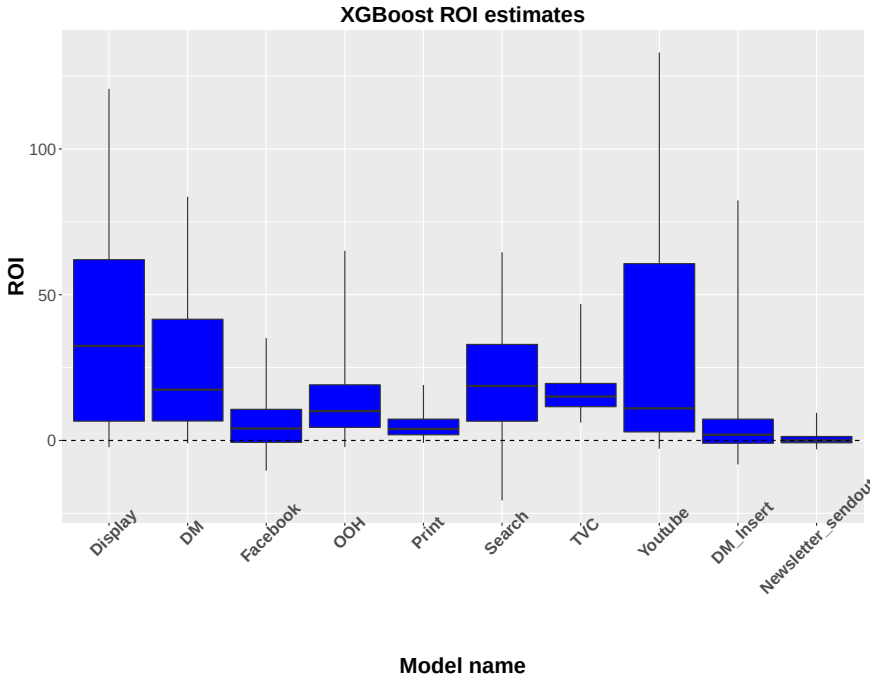


Figure 4.15: The ROI-estimates for XGBoost on the real data. The dashed line shows where 0 is located.

5 Discussion

5.1 Results

5.1.1 Ordinary least squares

The ordinary least squares, which to a large extent functioned as a baseline, displays mediocre results in terms of prediction performance in comparison to other models. This is consistent for both the simulated and the real-world data sets; especially for the RMSE. Sometimes the method ends up performing slightly better compared to the closely comparable methods, such as robust and constrained, and sometimes slightly worse.

An interesting observation, more prevalent in the real dataset than the simulated, is that the OLS has worse MAE than multiple models which it outperforms in RMSE. This suggests that the OLS adjusts more to the largest and smallest values than comparable methods. Shapley value regression can for example be seen to have a significantly larger RMSE but a significantly smaller MAE (see Table 4.4), and similar results can be seen for the robust method in the linear forms. The same effect is not as pronounced in the case of the other functional forms, although still present. The differences in the MAE are however a lot smaller for the log-linear and semi-logarithmic and could be due to random error.

In terms of coefficient estimation on the simulated data (see figures 4.4, 4.5 and 4.6), OLS provides quite wide confidence intervals for all functional forms compared to other models. Depending on which parameter estimate is studied this effect varies. For example, estimate for the `facebook...net` variable is quite large compared to the robust model in the log-linear case. However, the other way around is true for the linear case. As discussed in Section 2.2 the OLS has the BLUE property and therefore is the most accurate linear unbiased estimator in terms of variance. The results from the blockwise bootstrap suggest that it might be reasonable to instead use a biased method to get more certain estimates. The large confidence intervals are a problem in interpretation of the model since its effects cannot be consistently estimated. For example, the confidence intervals often include both negative and positive coefficient values, which have very different interpretations. The real-world data tells a similar story. The 95% confidence interval for the ROI of the `DM` variable spans from around −15 to 30, for example (Figure 4.13), indicating a high level of uncertainty. These carry wildly different interpretations. Where the case of −15 would be a large loss of money for each unit spent, the ROI of 30 could be considered a very high return even with any additional cost

not present in the media-budget. The width of the confidence intervals differ greatly between parameters and the functional forms. While the previous example shows a large confidence interval, the semi-logarithmic ROI for the `Facebook` variable (See Figure 4.14) shows a much smaller interval. This interval can, however, still be considered large as it spans from around 2 to 10 which still is a highly varying result.

In conclusion, the OLS works well as a baseline, performing decently in both predictive performance and on parameter certainty.

5.1.2 Generalised least squares

For the models where an ARMA-process was identified for the errors, using the GLS improved the result significantly in terms of quality of fit. As an example, the R^2-value went from 0.79 to 0.89 in the semi-logarithmic case by applying GLS, as seen in table 4.3. This indicates that a larger part of the variance in the sales is explained and a better quality of fit. It confirms the view that ARMA-processes can greatly improve the result if such a process exists and suggest that if such a process is identified, it should be incorporated into the model.

For the real data set, however, it is observed that the quality of fit, both in terms of error and R^2-value, is worse by applying GLS in comparison to OLS. Since the identified processes are $MA(1)$-processes with small coefficients, see Table 5.1, for both the log-linear and the semi-logarithmic models it seems a process was found in the data by random error. That is, an underlying process does not exist but one was found by chance. While including parameters not present in the underlying structure does not introduce any bias for the GLS, it does introduce larger variance in the predictions, since an additional estimated parameter is used. This in turn leads to worse prediction performance and is one explanation for the worse performance of the GLS compared to the OLS.

Simulated data

Model	AR	MA
Log linear	-	$(0.898, 0.968, 0.325)$
Semi-logarithmic	$(0.500, -0.202)$	$(0.312, 0.643)$

Real-world data

Model	AR	MA
Log linear	-	−0.164
Semi-logarithmic	-	−0.202

Table 5.1: Generalized least squares estimates of the $ARMA(p, q)$-process coefficients

When it comes to identification of the process order for the simulated data, the semi-logarithmic model succeeded in finding the right process, whereas the the log-linear model did not. This, in combination with the prediction metrics, suggests that a linear approximation of the logarithm of the sales can give competitive results when the underlying structure is log-linear. The true structure, the log-linear model, instead finds an $MA(3)$-process, not incorporating an AR-process of any order. While the true structure should be more accurate than a linear approximation, this can partially be explained by the following two factors. First, due to the duality of AR and MA processes, a stationary $AR(p)$-process can be represented by an infinite $MA(\infty)$-process [10]. It therefore seems likely that the $MA(3)$-process found for the log linear GLS is a good approximation to the $ARMA(2, 2)$-process added in the simulated data. Second, in the case where an $MA(3)$-process can reasonably represent an $ARMA(2, 2)$-process it might also be favoured by the AIC, which only penalises the model by the amount of parameters chosen for incorporation. It is possible that the AIC-based method for choice of order is biased downwards in such cases. For prediction accuracy, this should not result in a large difference. For example, the RMSE is reduced for both models. However, if

the interpretation of the ARMA-process is important, it is possible that other methods can provide an order more representative of the underlying model.

The linear method's lack of ARMA-process, by both visual inspection and AIC, is unsurprising in the simulated case. This is since the errors are not homoscedastic in the non-transformed space. Instead, since errors are added before exponentiation, the variance scales with the size of the sales. This in turn leads to the time-dependent errors being non-stationary and therefore does not work well with linear stationary time series analysis.

When it comes to the parameter estimates, the results are varying. The GLS does at times have tighter confidence interval compared to the OLS for the simulated data. However, their median can differ by quite a lot. For example, for the estimates of the `facebook...net` parameter (see Figure 4.5), the GLS confidence intervals are tighter and have a larger median for both the semi-logarithmic and log-linear. However, the opposite is true for the `search_branded...net` parameter (see Figure 4.6). Further, the ROI estimates follow a similar pattern, often delivering slightly tighter confidence intervals than the OLS along with a different median. This is however not significant in comparison to, for example, SVR (see discussion in Section 5.1.5).

The results on the real-world data show only very small differences between the OLS and GLS. Once again, a probable reason is that the $MA(q)$-coefficients are small and do not affect the model much. Again, it is likely that including an additional parameter increases the model complexity and slightly worsens the results. The same holds true for the ROI estimates in these cases.

The generalised least squares method has both theoretical and here empirical justification to be used over the OLS if the errors follow a linear stationary process. In this case, predictive performance increases in comparison to the OLS for the cases with a clear process, but coefficient certainty does not necessarily increase.

5.1.3 Constrained regression

The constrained models are showing the worst performance in terms of quality of fit for the linear and log-linear forms on the simulated data. These two models show the worst results for all four performance metrics for their respective functional form. One probable explanation for this can be due to the fact that these did not converge, meaning the algorithm did not arrive at an optimal solution. There are a couple of reasons why this could be the case. Mainly, it is confirmed that the assumption that the hessian is of full rank is not fulfilled.

However, in the case of the semi-logarithmic functional form on the simulated data, the constrained model instead performs the best in terms of RMSE and well on the other measures. In contrast to the other forms, the semi-logarithmic model did converge which could be the reason for this performance. It is possible that the other models would also perform well given that they converged. However, another possibility is that the restricted parameter space contains better estimations in the case of the semi-logarithmic form compared to the other two.

As can be seen in Table 5.2, many of the coefficient estimates get stuck on the lower bounds chosen for the predictors. This is expected when the non-constrained optimum lies outside of the constrained space. However, the problem of the interpretability still exists using this method and if the main goal of the modelling is interpretability, this method might not be preferred. On the other hand, the interpretation can still work out better. As an example, the `display...net` having no impact in the model might be better than a negative impact in interpretation. It is also interesting how other sets of boundaries affect the models. Assuming an expert knows the intervals which a parameter reasonably falls within, the parameter can still get stuck at these reasonable boundaries. This could prove useful for both interpretation and forecasting if these intervals include the real value, assuming there is one.

On the real world dataset the constrained models perform better on average, with results about equal to the OLS for all four quality of fit measures. One reason for this might be due

Name	Semi-logarithmic	Log-linear	Linear
`display...net`	**1.00e-10**	**1.00e-10**	**1.00e-10**
`facebook...net`	**1.00e-10**	4.73e-3	**1.00e-10**
`search_branded...net`	5.41e-6	4.18e-1	**1.00e-10**
Converged	Yes	No	No

Table 5.2: Coefficient estimates for the media predictors for the constrained models on the simulated data.

Dataset	$\mathbf{X}$	$\mathbf{X}_{Log-linear}$
Real data	$2.97 \cdot 10^8$ (singular)	130 (non-singular)
Simulated data	$4.58 \cdot 10^8$ (singular)	312 (non-singular)

Table 5.3: κ-values (the conditional numbers) for the data matrices $\mathbf{X}$ (used for semi-logarithmic and linear models) and $\mathbf{X}_{Log-linear}$ on the simulated and the real-world data set respectively. The higher number, the more ill-conditioned the matrix is, leading to the matrix being closer to computationally singular.

to fewer negative parameters in the optimal solution, making the optimisation problem close to an OLS. Still, none of the three models converged for the real-world dataset. This is a further indication of problems with the optimisation method, but the effects do not seem to be as severe as for the simulated case.

When it comes to the confidence intervals of the parameter estimates, the results differ between the simulated and the real-world datasets. For the simulated data, the confidence intervals are in many cases too wide to, in an informative way, show in the same plots as the other methods. This might be due to convergence problems when optimising the model on the bootstrap samples. Regardless of the reason, the exact model tested is not consistent for the parameter intervals. The real-world dataset tells a different story and the constrained model often performs similarly to the OLS. While differences do occur, they are in most cases quite small. For example, the confidence intervals for the `TVC` parameter are slightly smaller for the linear form and larger for the log-linear form compared to the OLS. Both of these cases also have similar medians to the OLS confidence intervals. The largest differences can be seen when a large part of the OLS intervals is negative and the constrained instead is forced positive. The linear form of the `Facebook` parameter is an example of this, see Figure 4.12.

The ROI estimates instead show mixed results. The estimates for the log-linear form are in multiple cases very wide compared to the other models, indicating very uncertain estimates. An example of this can be seen in Figure 4.13, where the constrained log-linear model's confidence intervals span from about 0 to 200. For the other functional forms, the linear and semi-logarithmic, the ROI estimates are more consistent with the OLS. For example for the `DM` variable (see Figure 4.13), the linear constrained model's confidence intervals look similar to a OLS confidence interval cut off at 0. For the same parameter the semi-logarithmic constrained model's CI is shifted upward and has a slightly tighter 95% CI compared to the OLS.

As mentioned previously, the underlying issue in the simulated data is that the first assumption, that $\mathbf{X}$ should be of full rank, was violated for the linear and the semi-logarithmic functional forms. As seen in Table 3.2, the media variables suffer from strong multicollinearity, which only occurs with a singular matrix, leading to the hessian $\mathbf{X}^T\mathbf{X}$ being singular or nearly singular. How the model would perform in the converged case is hard to say since it is unclear how close to the optimum the non-converged solutions were. With multicollinearity often being an issue within MMM as discussed by Hanssens [41], the Gauss-newton method might be deemed unsuitable, and other methods that can enforce restrictions on the coefficients that does not need this assumption to be fulfilled should rather be used.

On the real-world data for the semi-logarithmic and the linear functional forms this also seems to be the issue, as the data matrix $\mathbf{X}$ was in fact computationally singular. Here, the

algorithms did not converge, but the constrained models seem to be performing on par with the OLS for the linear and semi-logarithmic forms, both in terms of quality of fit and coefficient estimation. The conversion to logarithmic values did however make the data matrix $\mathbf{X}_{Log-linear}$ non-singular, but the Gauss-Newton method did still not reach a converged solution. An explanation for this might be that the default of starting values were not suitable. However, the condition number of $\mathbf{X}_{Log-linear}$, cf. [21], is still relatively large at 130, and it can be suggested that $\mathbf{X}_{Log-linear}$ is nearly singular (see Table 5.3).

If the enforced parameter interpretation can still be of value, i.e., that it is acceptable that the coefficient obtains a value on its boundary, the method can prove useful. However, if it is not, there are other methods which perform better on both uncertainty and prediction performance. The log-linear constrained model follows the same reasoning, however, the ROI estimates seem to be problematic in combination with this.

Name	**Semi-logarithmic**	**Log-linear**	**Linear**
`Display`	1.70e-07	**1e-10**	40.6
`DM`	7.16e-08	0.065	8.25
`Facebook`	1.42e-09	0.0065	**1e-10**
`OOH`	6.81e-08	**1e-10**	5.06
`Print`	8.18e-08	0.125	14.5
`Search`	7.03e-08	0.096	22.35
`TVC`	1.15e-07	0.026	9.21
`Youtube`	1.85e-07	**1e-10**	10.34
`DM_Insert`	1.43e-07	0.005	27.81
`Newsletter_sendouts`	7.46e-08	0.048	14.49
Converged	No	No	No

Table 5.4: The coefficient estimates of the models on the real data.

In short, the constrained methods show some promise and could be useful given correct boundaries and optimisation. However, as implemented here, the constrained models generally perform poorly for both predictive performance and parameter certainty. Further, another optimisation method should be used instead of Gauss-Newton as MMM data tends to have non-singular data matrices.

5.1.4 Robust regression

The robust models perform, on both the simulated data and the real data, worse or equal to the OLS. Since the data is simulated with normally distributed errors, the distribution is not heavy-tailed and there are no real outliers; although 'extreme' points which look like outliers are possible. As a result, the ordinary least squares is expected to perform better as the reasons to use robust methods are not present.

One explanation for the cases where the robust method performs equally to the OLS is that the number of large errors is very small. In such cases, the robust method effectively works in a similar way to an OLS. As seen in Section 2.7.2, the MM-builds on the Huber estimator which up to a cut-off uses the squared distance loss function and using a scaled linear distance for points further away. If all of the data points fall within this interval or close to it when fitting, the method will function similarly to the OLS. This seems to be the case for the simulated data where the robust and OLS can be seen to have very similar results. As previously stated, the errors are heteroscedastic on the simulated data in the non-transformed space. Despite this, the linear OLS fit to the simulated data still shows residual quantiles close to those of a normal distribution, as can be seen in Figure 5.1. This could be one reason the robust does not outperform the OLS even in the linear case.

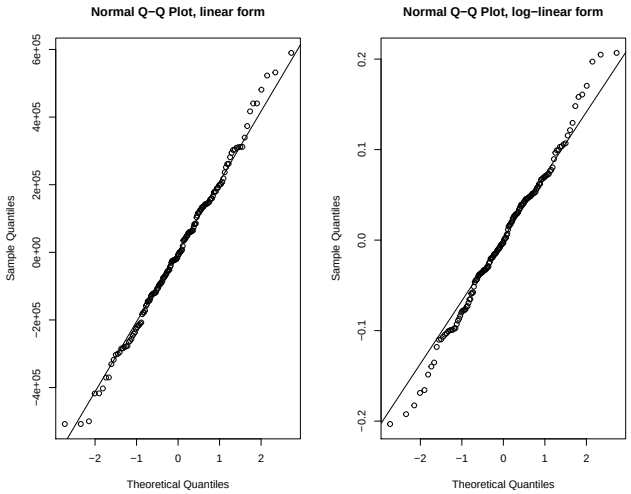


Figure 5.1: A Quantile-Quantile plot of the residuals form the linear and log-linear OLS on the simulated data.

Table 5.5: The reduction of MAPE when using the robust model on the real data

Real-world data			
Model	Linear	Log-linear	Semi-logarithmic
OLS	26.12 %	15.03 %	14.81 %
Robust	20.08 %	14.66 %	14.37 %
%-unit reduction	6.04 %	0.37 %	0.44 %

On the real world data, this does not seem to be present to the same extent. The robust models instead perform worse on RMSE for all functional forms. However, the robust methods generally perform about equally or better on the MAE and the MAPE. Especially the linear robust model, which performs a lot worse than the OLS on RMSE but a lot better on the MAE. This indicates the robust is more accurate when the errors are small rather than when they are on the larger side whereas the OLS could be more balanced. This can also be seen in Table 5.5, where the MAPE is smaller for the robust model compared to the OLS for the real data.

On the simulated data, the confidence intervals for the robust models are quite wide, often proving worse results than the OLS. This is an indication that the method falls behind the OLS in this measure, if there are no outliers. While this is true for the linear and semi-logarithmic forms, there are cases where the robust method performs better in the log-linear case. The log-linear estimates for the `facebook...net` parameter (see Figure 4.5) and the `search_branded...net` (see Figure 4.6) show tighter confidence intervals compared to the log-linear OLS. The last variable, `display..net`, also has a tighter 95% confidence interval, but the 25% and 75% quantiles are further apart. This is especially interesting since the log-linear form is the structure the data is simulated after and the lack of outliers should make the OLS more certain. The exact reason for this is not identified and it is possible that it is just restricted to the media-variables and not the other.

When it comes to the real world dataset, the robust generally performs about equally to the OLS in parameter certainty. Some parameters estimates show smaller confidence intervals for the robust and some show smaller for the OLS. No large differences between these seem to be present in terms of the parameter estimates.

To summarise shortly, the main benefit of the robust method is a lower MAE and MAPE when outliers are present. Other than that, no real benefits of robust regression, in comparison

to OLS, is shown in the context of these datasets. It is still possible that reasons to use robust methods exist in other datasets and a residual analysis in the future is suggested to identify or reject such cases.

5.1.5 Shapley Value Regression

The Shapley value regression (SVR) is an interesting case in many aspects. While the SVR performs slightly worse than other models, such as OLS and GLS on the simulated data, the uncertainty in parameter estimates is lower. It has the smallest confidence interval for all three media parameter estimates on the simulated data.

The most interesting case is the log-linear functional form, since the data was simulated by assuming such a relation. As a result, the true parameters are known for the model which makes it possible to evaluate the accuracy of parameter estimation. As mentioned, the confidence intervals are very small for the parameter estimate and in two of three cases the true values is included in the 95% confidence intervals. In the last case, the `display...net` coefficient estimate (see Figure 4.7), this is not the case. From these results, it is possible that the SVR is biased and on average overestimates this variable. This can be compared to the OLS and GLS which for the correct structure are known to be unbiased. However, these models have much wider confidence intervals in general, and the 50% confidence intervals are in most cases wider than the 95% intervals for the Shapley value regression. If accuracy of parameter estimate is important, the SVR might therefore still be a good alternative due to its low estimator variance even if it is biased.

The linear SVR is also very interesting in the case of the ROI estimates. For the linear model the ROI and parameter estimates are equal and the linear models can be seen as a direct estimate of the linearised ROI. Since the true structure and parameters are known, the true linearised ROI is also known and this direct estimation can be evaluated. It can be seen in Figure 4.7 that the linear SVR performs very well in estimating this linearized ROI. In two out of three cases, the confidence intervals for the linearised ROI are spot on. The last case is a slight underestimation which still is very close. This can be compared to the semi-logarithmic and log-linear SVR models. While the semi-logarithmic SVR has slightly tighter confidence intervals, it consistently underestimates the linearised ROI in the case of the simulated data. In the three cases of this data, the semi-logarithmic SVR estimates the linearised ROI to be about half, or less, of the actual effect. This is rather interesting since underestimating the effect could be beneficial in certain situations, such as optimising investments, compared to overestimating the effect. The log-linear SVR instead overestimates the linearised ROI to such an extent that it is not informative to show its confidence intervals in the same plot as the semi-logarithmic and linear model. These results in combination suggest that it is possible that it is more efficient to use the direct approach of linear SVR for the linearised ROI, rather than extracting it from a non-linear model.

In the case of the real data, the SVR, just like for the simulated data, has a slightly worse fit for all three functional forms. The uncertainty in parameter estimates is, again consistent with the simulated data, a lot lower compared to most other models. In almost all of the cases, the confidence intervals for these parameter estimates are considerably smaller compared to the OLS. However, in the case of the `TVC` parameter estimate (see Figure 4.11), the SVR has a larger interval compared to the OLS for the log-linear model. The results therefore indicate that the uncertainty in the parameter estimates is in most cases reduced by applying SVR. The `TVC` estimate does however prove that this is not always the case (see Figure 4.11).

When it comes to the ROI estimates, the SVR once again has very tight confidence intervals compared to most other models. For example, in the case of the `DM` parameter, see Figure 4.13, the SVR shows significantly smaller confidence intervals compared to all models except the hierarchical varieties. This is the case for all three functional forms, all showing tighter confidence intervals for most parameters.

Some interesting behaviour where the SVR estimates the effect very differently compared to the OLS can be seen in some cases of the ROI estimates. For example, the SVR has a high ROI value for the `DM` parameter for both the linear and the semi-logarithmic forms. These estimates are for both functional forms a lot larger compared to their OLS counterparts. The SVR estimate for the semi-logarithmic is for example far outside of the 95% confidence interval of the semi-logarithmic OLS.

In conclusion, the Shapley value regression performs very well in parameter certainty, but in turn offers some quality of fit.

5.1.6 Bayesian structural time series

Whereas the SVR is rather easily explained as to why it works well, the BSTS models, in particular in the linear form, are more surprising for a few reasons. First of all, it does not follow the correct functional form of the simulated data (being log-linear).

Looking at the RMSE on the simulated data (Figure 4.2), all BSTS models have a better result than most other models of each model's corresponding functional forms, indicating they might have an advantage against other models. This is however not the case for the real data (Figure 4.9) where the log-linear and the semi-logarithmic BSTS models perform poorly.

Looking at the coefficient and ROI-estimates, the BSTS models do not perform particularly well in comparison to others. Many times, the BSTS models' coefficient and ROI estimates were excluded (see Figures 4.13, 4.14) and the estimates are often very insecure, many times having confidence intervals being wider than other models (see figures 4.4, 4.5 and 4.6 for examples).

First of all, one thing to question, is the amount of samples used: only 4,500. Although this is in accordance with the original paper by Scott et al. [51] that used 5,000 samples, Gelman [38] promotes that around 2,500 samples are needed for an accuracy of 1 %. Thus, a sample size of 3,500 (having a burn-in of 1,000) can be considered sufficient. Given that more than 2,500 samples were used for both the linear and the semi-logarithmic models, and the models still display similar behaviour, this does not seem to be the issue for neither the real or the simulated data.

Second of all, the way ROI-estimates are calculated for the semi-logarithmic and the log-linear models includes a normalisation of the contributions, which distributes the contributions over the regressors. If the model is sparse, including only a few variables, which is the intention of the BSTS model originally, each regressor will in each draw get more contribution distributed on itself, creating an inaccurate estimate of its actual contribution and therefore the ROI. This thus indicates that the method of calculating the ROI-estimates for the semi-logarithmic and the log-linear functional forms might not be suitable for the BSTS model. However, this does not explain the relatively wide confidence intervals of the coefficient estimates.

Looking at the coefficient estimates for the log-linear BSTS on the simulated data (see figures 4.4, 4.5 and 4.6), it seems that although the confidence intervals are wide, this is the second best at getting its mean close to the true value after SVR. This suggests, although it has a high uncertainty, that it can be correct in terms of coefficient estimates and potentially mitigate multi-collinearity, which could be due to the sparsity-property of only including a few variables in each MCMC-draw. However, it can still be quite uncertain due to having wide intervals.

One of the differences for the BSTS methods is that they use another way of modelling the seasonality. This also seems to be the reason why an $AR(4)$-process is found for the linear BSTS model, while none is found for the regular OLS model. This way of modelling seasonality might be an explanation for the linear BSTS model's great performance. However, if this were to be the sole reason of better estimates, the other functional forms would display better results as well.

One possible explanation for the prediction accuracy of the linear BSTS is that it is not extremely accurate, but has a very low variance in prediction. Such a case can result in better

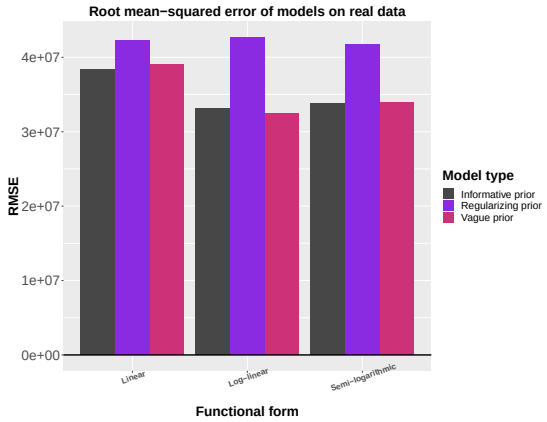


Figure 5.2: The RMSE for the different hierarchical models.

prediction performance while giving biased predictions. While this could be the case for these predictions, this does not seem to be the case when examining individual confidence intervals of parameters. These confidence intervals do not indicate that a low variance is present in any form. However, due to multicollinearity it is possible to have predictions with very low variance, while the same is not true for individual coefficients. As a result, the confidence intervals are not enough to rule this out, and further investigations would be needed to understand exactly why the linear BSTS model is performing this well in terms of prediction.

In summary, it seems that the linear BSTS model would be an excellent candidate for prediction performance, but perhaps less so for obtaining confident coefficient estimates. One might further explore whether more confident estimates are obtainable by having a larger sample size; this was however not done due to time-restrictions, and it was not deemed credible as the theory suggests otherwise [38].

5.1.7 Hierarchical Bayesian regression

The hierarchical model performs well on quality of fit for all functional forms, beaten only by the linear BSTS. Both the vague and the informative prior show very similar results in terms of both RMSE, MAE and MAPE for all three functional forms, while the regularising priors shows weaker results in general. Since the informative and regularising priors where implemented with the same strength, this might indicate that the effect of the media channels are rather similar across the three countries considered.

The good results of the hierarchical model indicate that modelling on a store-basis can be beneficial. This increased prediction performance can depend on a few things. First of all, it is possible that the variance of predictions on a store-basis is simply lower when summed up, compared to the on a country level. The coefficients can be estimated more precisely on a store level which improves the prediction performance on a country level. If this is the case, estimating with an OLS for each store can also provide good results.

Another explanation is that the variance of the estimation of the parameters is much lower due to the way of modelling the similarities between the store coefficients. Through modelling a mean of the store coefficients and to which extent these should be 'pulled' towards it, the variance of the estimates is lowered. This in turn reduces the prediction error. This does, however, introduce some bias which instead increases the prediction error, and once again it is a trade-off between bias and variance. Since this effect is present with all three priors, the results would have to be compared with an equivalent unbiased model in order to know if the benefits from the reduced variance counteracts the bias.

A last possibility, in the case of log-linear and semi logarithmic, is an increase in performance due to the differences in relation between the store- and country-level models. The

total sales of a country can in the linear case be represented as

$$\hat{y}_i = \sum_j \hat{y}_{i,j} = \sum_j X_i^T \beta_j = X_i^T \underbrace{\sum_j \beta_j}_{=\bar{\beta}} = X_i^T \bar{\beta} \tag{5.1}$$

which is a model equal to a country level linear model. However, the same is not true for the log-linear or semi logarithmic models where the structures assumed on the store level can not be aggregated into an equal model on the country level. Since the hierarchical models follow a different structure this could also be a reason why they could perform well. However, since the models perform better in all three cases of functional forms, this cannot be the sole reason for the improvement throughout all cases.

The ROI estimates are very tight for the hierarchical models, competing even with the Shapley values regression many times, for example seen in Figure 4.13. There are a few reasons which could explain this behaviour, many of which already stated. At least in the linear case, this stems from a lower variance in estimation of the parameters. However, this is a plausible explanation for the tight confidence intervals of the other two functional forms as well. This decreased variance can by explained by the previously mentioned lower store-level variance or similarities between the stores. A combination of both is also possible. Regardless if it is the first, second or both, the variance is lowered and therefore also the ROI in most cases. It is however not always the case; in the case of the `Facebook` variable (see Figure 4.14) where the confidence intervals, in the semi-logarithmic case, are wider than for many of the other models, including OLS. Further, note that the theoretical nature of these confidence intervals, sampled from an assumed distribution structure, could make these tighter than they should be.

Looking at the linearised ROI estimates, there are a few clear patterns. First, the informative prior changes the estimate by quite a bit compared to the vague case, in many cases. For example, in the case of the linear form `Facebook` ROI estimate (See Figure 4.14), the informative prior has its median around -5, whereas the vague prior has its median closer to -70. Similar results for the semi-logarithmic and log-linear case can be seen for the `Facebook` parameter. Another example of this, with a smaller effect, is the linear form's ROI estimate of `DM` (see Figure 4.13), where the estimates fall just below 30 for the vague case and just above 40 for the informative.

The regularising method, on the other hand, does not seem to change the ROI estimates to the same degree, as can be seen in both the previous examples. This prior instead has its own interesting behaviour. One of these behaviours is that the model with the regularising prior sometimes predicts a higher ROI compared to the vague prior for the linear case. Examples of this include the `DM` ROI as seen in Figure 4.13. The parameters in this linear case can be seen as the ROI on a store basis and the ROI can therefore be seen as an aggregation of the parameters. The regularising aspect will in general tend to be 'pulling' these parameters closer towards $\mathbf{0}$ compared to the vague model and should in most cases also pull the ROI estimates towards $\mathbf{0}$. This is present in some cases and can be seen in a slight manner for the `Facebook` ROI estimate, however, not for all.

One possible issue with the hierarchical models are their sometimes unreasonable ROI estimates. While these estimates are certain, and they in several cases provided realistic estimates (see Figure 4.13), they can sometimes be regarded as unrealistic. For example, the `Facebook` ROI estimates (See Figure 4.14) of the vague and regularising priors fall around -40 for the semi-logarithmic form and around -70 for the linear form. It can probably be argued that increasing Facebook advertisement spend by one unit of money is unlikely to lower the sales by 70 of that unit. Assuming it is unrealistic, the hierarchical models do not always perform well in identifying a 'correct' ROI.

In summary, a hierarchical model perform very well in terms of prediction accuracy and certainty on the ROI-estimates, beating all other models except SVR in most cases. However,

it must be noted that the ROI estimates of the model sometimes were among the highest or lowest, therefore possibly over- or underestimating these estimates.

5.1.8 XGBoost

Since the XGBoost estimates a function from a space of functions, it allows for very accurate representations of a problem given that this space is large enough. In this case, however, due to the small size of the dataset the method can easily be over-fitted to the training data if this function space is too large. The results on the simulated data indicate that through enough regularisation, this function space seems to be restricted to a space which contains a reasonable representation of the relation between the response and predictors without completely over-fitting to the training data. This also leads into the larger discussion of regularisation. The XGBoost is the only method, apart from the hierarchical methods, which uses regularisation to restrict the model. This regularisation usually has the effect of lowering the variance and increasing the bias by restriction of the function space. Finding the balance between these can increase the prediction performance as explained by the bias-variance decomposition, see section 2.10.1. By applying regularisation to other modelling methods it is possible that these could match the XGBoost in terms of prediction performance. For example, Mhitarean-Cuvsinov [59] compared regularisation methods in a MMM setting and found that regularisation methods such as ridge regression and LASSO outperformed the OLS by a considerable amount in prediction performance.

On the real data, however, the XGBoost provides a worse generalisation error. This goes against previous results and beliefs that the method would prove flexible enough to perform well in such a case. Why this occurs is not certain and could have many explanations. One explanation is that it provides by far the lowest error on the training set on both datasets, which indicates a significant overfit to the training data. This despite choosing the parameters based on a random-search cross-validation (RS-CV) with model selection on the validation-set RMSE. It might be possible that there are a set of parameters which make the model perform equally to or better than the well-performing models. Due to the randomness of the random search cross-validation, many parameter combination go without testing and it possible that it missed such a combination. However, since the parameter search was run with 10,000 iterations, it seems likely that a set of parameters close to an 'optimal' has been tested; in particular considering that randomized searches has been found to provide similar results to exhaustive parameter searches, as described by Bergstra et al. [4].

Another explanation is the randomness of the XGBoost. Randomness is introduced when training the model and running the same experiment twice with XGBoost do not necessarily end with the same result if the random generator is not controlled. Although it was to some extent controlled, the package for `RandomizedSearchCV` only allowed a certain guarantee of running the experiments deterministically, as described in the documentation[1]. It is therefore possible that the chosen parameters in most cases will give a significant overfit, however, on the specific run also happened to provide a good fit to the validation data. Bishop [6] mentions that if many iterations are run in the model selection, there is a possibility to overfit to the validation data. Given that 10,000 iterations were run, this seems plausible. A method to counteract this would be to in some way penalise the model-complexity in the model selection of RS-CV as well. Similar to an information criterion, such as AIC, the maximum tree-depth and the maximum amount of trees could for example be penalised. A set of parameters allowing for a complex model could then be rejected in favour of a more restrictive set if they have similar RMSE.

A third possibility is that the parameter search was done over a range of too many possible hyper parameter choices, risking an overfit despite the randomised search to minimise the

[1] Documentation available at `https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.RandomizedSearchCV.html`. Accessed 18/6 - 2019.

cross-validation error. One might further explore whether stricter choices of hyper parameters might lead to a simpler, more general model.

Simulated data		**Real-world data**	
Model	Train/test RMSE ratio	Model	Train/test RMSE ratio
Semi-logarithmic Shapley	0.594	Log-linear Shapley	0.792
Linear BSTS	0.102	Log-linear BSTS	0.097
XGBoost	$4.23 \cdot 10^{-4}$	Semi-logarithmic	$1.61 \cdot 10^{-7}$

Table 5.6: Ratios between the training and the test RMSE, comparing the highest and the lowest ratios on the parametric models with XGBoost on both data sets.

For the simulated data, the ROI estimates can be seen to be relatively fair, being comparable to other models. However, the estimates are very uncertain on the real-world data set. There could be many reasons for this uncertainty. First of all, the SHAP values do not give the full picture of the function estimated by the XGBoost and can instead be seen as a simplification. The linear approximation is then run on this simplification, and can therefore be seen as a simplification of a simplification. It is possible that the information lost in the steps of this method contributes to the large confidence intervals.

Another explanation is, once again, the model complexity. Due to the overfit present in the models it is not surprising that this effect varies largely. The large subset of functions could contain many well-fitting functions with very different SHAP values and therefore linearised ROIs. Like for the RMSE, the possibility exists that the ROI estimates could benefit from having a larger regularisation. As seen in table 5.6, XGBoost has a significantly lower ratio between the test set RMSE and the training set RMSE, in particular on the real dataset, indicating an overfit especially on the real data.

Studying the plots of the SHAP-values for the media coefficients (Figure 5.3) of the simulated data, one can see that `facebook...net` follows somewhat of a concave response curve, which goes along with the structure of the simulated data, as it was formed using the log-linear functional form. However, the SHAP-values for `display...net` rather starts to decrease after a certain value, and the `search_branded...net` appears to have somewhat of a linear relationship. This indicates that the SHAP-values could relatively accurately capture the true relationship between the response variable and the predictors, but further research is needed to confirm this.

It is also possible that the XGBoost is not a very good model for use in conjunction with the linearised ROI. Thus, it might be interesting to try another non-parametric regression model, but with the same method.

In summary, the results indicates that a less overfitted, non-parametric regression model could give better results with this specific method, but further research is needed to confirm this. The method used for calculating the ROI estimates can not be dismissed nor confirmed to perform well. The results indicate that the method could potentially perform well, given a less overfitted, more general model.

5.1.9 Comparing functional forms

Studying the differences between the functional forms, it can be seen out of a general perspective that the semi-logarithmic and the log-linear functions both perform better in general on the quality of fit in terms of RMSE (see Figure 4.9 and Figure 4.2), with a few exceptions.

Regarding uncertainty, the functional forms can rather be compared in ROI than in coefficient estimates, as these are on different dimensions. Using the methods provided in this thesis, it seems as if the log-linear method tends to overestimate the ROI, while the semi-logarithmic and linear both provide more reasonable estimates.

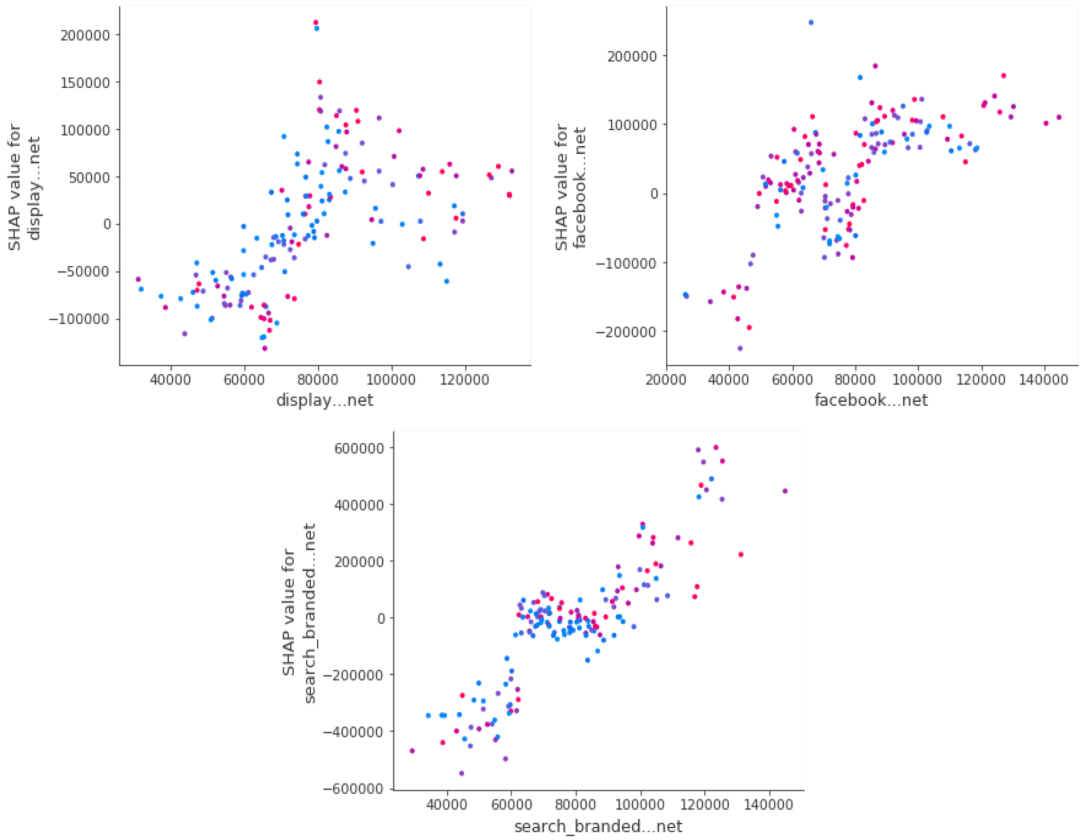


Figure 5.3: The SHAP values for the media variables for the simulated data. The colours of each point corresponds the value of another predictor and does not have a meaning in this case.

5.2 Method critique

While the goals of the methods chosen were to provide reliable and comparable results, there are always flaws and several discussion points where the method could be critiqued. Here follows the main critique identified by the authors.

5.2.1 Methods of interpretation

It is important to note that the main task in MMM is to understand the relations between advertisement and spend. This can for example be done through derivatives and inspection, however, a simplification can be helpful and one such simplification is through ROI-estimates on the different marketing channels. There is no definite, golden standard answer for these. Thus, one must be critical in the assessment of the models, regarding the uncertainty. One can never know the exact value of the actual parameters as an economical system can be regarded as chaotic systems of the second order, meaning that the outcome might change if one predicts the values [42].

The way the ROI-estimates are calculated here is also a large simplification, made to compare different functional forms, and should not be considered a definite truth. How to do this well should be further researched in order to provide a good and representative method. Additionally, the fact that all log-linear models showed unreasonable ROI-estimates indicates that there is an error in the way it is calculated, rather than it actually providing unreliable ROI-estimates. Further, a question is whether the different forms of evaluating the ROI-

estimates used measures the same thing. Theoretically, they should provide the same ROI-measures, but as seen in the results, the results for the log-linear models differed greatly from the linear and the semi-logarithmic models, indicating that the method might not be adequate in the log-linear case.

5.2.2 Comparing functional forms

Studying the differences between the functional forms, it can be seen out of a general perspective that the semi-logarithmic and the log-linear functions both perform better in general on the quality of fit in terms of RMSE (see Figure 4.9 and Figure 4.2), with a few exceptions. However, the R^2-value is significantly higher for the linear models (see Figures 4.1 and 4.8). This discrepancy that contradicts what the other measures of quality of fit indicate could be explained by the fact that the R^2-value was always measured in the linear setting. The other measures (MAE, MAPE and RMSE) were measured and compared to the actual response variable y_t, which only the linear functional form had as response variable in a linear setting. As a result, the R^2 can be seen as the percentage of variance in the linear space for the linear OLS and log-transformed space in the semi-logarithmic and log-linear space. Therefore, there are issues in the direct comparison of these models. Due to this, the discussion regarding the results has mainly revolved around the other measures and not the R^2-value.

5.2.3 Causal modelling

All the models attempted in this are regression models and cannot without further adjustments be considered causal models, as discussed in Section 2.1.4. Thus, one must strongly critique the method in the essence that one is rather attempting to obtain causal relations out of a non-causal model. However, despite relying on causal effects to properly answer the advertisers questions, there is a lack of research in applying existing methods of causal inference within the MMM context. One explanation is for this is knowledge of such methods. Pearl [61] mentions that while methods and algorithmic tools for causal inference are developed, there is a lack of knowledge of these by researchers who could put them to practical use. On the other hand there are also difficulties in causal inference. Peters et. al. claim that *"learning causal structures from data is only doable in rather limited situations"* [62]. Further, Chan et al. [19] explains why causal models such as randomised experiments and potential outcomes are infeasible in an MMM context, and why one turns to regression models instead. Due to these difficulties, the authors of this study focused on investigating models within the current framework that MMM offers, rather than investigating causal models in the context.

5.2.4 Variable selection

In this work, choosing variables through the use of statistical methods was not a focus. On the simulated data set, 10 out of 16 variables that do affect the result were chosen since this provides an imperfect choice of variables and information. This is often the case, as one in a practical situation often is missing some variables that have true effects, and also have other information having true effect that might, by mistake, seem intuitive to have effect. An alternative strategy would have been to have taken the variables that truly do have an effect. However, the study would then assume the condition of perfect information, which is something that rarely occurs in practice, as discussed by Quandt [63], who means that the data is rarely complete, and often lacking in detail.

5.2.5 Real-world data set

An important restriction is that the study studied a single real-world data set. This has the main issue that the findings of this study could be very specific to that specific company. As discussed in the previous section, the structure of the sales data might vary greatly due to

macro- and micro-economical factors, which creates a wide variety of sales- and marketing patterns, which might make other models be more well-suited. As a result, we encourage research challenging these findings in order to give a more complete picture.

Further, the simple imputation of the real data does bring an error, as the imputation method chosen as discussed previously in section 3.2.1, does create a worse result than other imputation methods [5]. Although this is not an issue in the results obtained from the simulated data, it does contribute to an error in the real-world data which is difficult to estimate. For future work, more sophisticated imputation methods are recommended and preferred. However, as the imputation is done in the same way for all models, it does not affect the comparison between the methods applied to the same data.

Lastly, the data set provided by Nepa is not and will not be made available to the public. The study will therefore not be directly replicable, but experiments are however still replicable with similar data sets.

5.2.6 Simulated data

While the simulated data is useful for evaluating the models due to knowing the true underlying structure, it might not suffice as a representation of a real dataset. The structure is not completely consistent between the simulated and the real-world datasets when studying them graphically (see figures 3.1 and 3.2) and as a result, the simulated data visually does not seem to be a complete accurate representation of the latter. However, the data was generated according to what the theory proposes, as it has the log-linear functional form, providing the most likely response curve for media variables as claimed by Hanssens [41]. The reason it is used regardless is that it is one way to know the true structure of parameters as parameter estimation is an important part of the study, as well as ensuring multicollinearity for the media predictors. Both of these factors were not abstracted away, but rather carefully chosen and saved for analysis.

The generation of data can be criticised in several ways. First of all, comparing the two figures, the generated data looks very different in its structure from the actual data. To begin with, although a seasonal pattern is clear in both datasets (see figure 3.1 and 3.2), the relative fluctuations in the simulated data are greater, indicating an exaggeration of the time-dependency property of the data. However, whether these fluctuations in sales are realistic have not been confirmed nor denied, as there exists many industries and companies with different sales patterns.

Another critique is that the simulated data was not generated with a hierarchical structure. In other words, data was not generated on store basis, but instead on a national basis due to the complex nature of generating data on a store basis, which did not allow for the hierarchical models to be modelled on the simulated data. This does make the results of the hierarchical models less reliable than, for example, the SVR results. However, the real-world dataset is more important than the simulated data set, and the results themselves on the data-set can still be considered valid.

5.2.7 Modelling issues

It must be addressed that the breadth of the work limited the possibility of going deeper into some models, which might have been of interest. Due to time-constraints, complete models from packages were in general used rather than building full models. It would, for example, be possible to build a complete Bayesian Structural Time Series model specified for this specific purpose with other tools such as `Stan`[2], which would allow for more freedom in the choice of priors and modelling. However, to mainly use pre-made packages in R and Python for the experiments has advantages as well. It reduced the time taken to construct the models

[2] `https://mc-stan.org/`. Accessed 18/6 - 2019.

and strengthens the reliability as the packages are (most likely) thoroughly tested and used by many before. Furthermore, it further strengthens the replicability of the study, as this enables other people to reproduce the same results by simply using the same functions in future studies.

5.2.7.1 Bootstrap

Bootstrap sampling was for all models but XGBoost performed with blockwise bootstrap in the same manner. However, during some bootstrap samples errors occurred as a result of the model not finding a solution. The reason for this was, in all observed cases, that the data matrix was computationally singular, which some of the methods cannot handle. In such cases the bootstrap rejected these samples and drew a new sample. As a result, the method could to some extent favour methods that could not handle such samples as these are likely to provide the least certain estimates. However, the confidence intervals of the methods affected (the constrained, robust, GLS and SVR models) do not vary from their expected values compared to the OLS. A solution to this problem, which is potentially more fair, is to draw the same samples for all models and reject samples for all models which are rejected for one. This might change the confidence intervals, but considers the models on the same basis.

5.2.8 Replicability

In this study, all models except for the hierarchical models, BSTS models and XGBoost were models not using any type of randomisation, and are thus highly replicable for the simulated data set by simply copying the method as described in the method chapter. For the models incorporating a random element, seeds were used to ensure replicability. Thus, if one wishes to reproduce the results using the simulated data, one can refer to Table B.2 (see Appendix B) for the seeds that were used. The seeds should not be of utmost importance in the BSTS and hierarchical model cases, as these will, given a sufficient amount of samples, converge towards the same distributions. The amount of samples used in both of these experiments should, according to the theory, be sufficient. However, the seeds can prove important in XGBoost, as the construction of the model involves randomly generating trees, making the seed important to generate the exact same model. It was also important to use when generating the simulated data, to ensure that the same data is generated.

5.2.9 Source criticism

During the pre-study, a number of papers were examined along with the main sources as mentioned in section 3.1. While papers performing research on Marketing Mix Modelling do exist, and research has been done previously, the area is seemingly not widely researched in an academical context. As a result, finding peer-reviewed sources proved to be difficult although some were found [53, 8, 70, 28, 76]. Thus, some of the domain specific knowledge for MMM was derived from technical reports from Google [48, 69, 19] or other white papers [27], which at times cannot be considered equally credible as these have not been peer-reviewed. However, the main sources used [44, 6, 38, 66, 41, 64] can be considered very credible, being published books with many previous citations, often with +1000 citations.

5.3 The work in a wider context - is marketing ethical?

There are of course ethical and social aspects in this work that should be addressed. Performing analyses in order to maximize the sales is something done in many areas, but is of course questionable ethically as it contributes to an increased consumption. One might also question marketing out of a democratic perspective, and there are different claims in this area.

Wejryd [74] claims that a high level of consumer choice, which a high level of marketing promotes, undermines democracy as it holds back political engagement by creating a passiveness among the citizens. On the other hand, Jocz et al. [49] claims on the other hand that marketing is democratic due to the fact that it expands the consumer choice, enabling a more varied consumption. Thus, by informing people on the choices they have, one increases their choices and expands their possibilities.

It is clear that marketing expands consumer choice by enlightening the consumer about different products that exists. However, this requires marketing under fair conditions where the customer is reached equally by all different choices, and presented equally to them. For the information sent out through the marketing to be beneficial, this information must be balanced and truthful. One of the goals with Marketing Mix Modelling and many other marketing optimisation strategies is, more or less, to overcome this and give reach out to the customer in a higher extent than the competitors, which might lead one to question whether one strives for the best possible outcome for the consumer rather or for the company.

6 Conclusion

In this study, a number of different models with different properties were modelled to examine which one should be used in an MMM context based on two evaluation factors: its certainty in its estimations of the coefficients and ROI-estimates, as well as their quality of fit.

Uncertainty can be mitigated in many different ways. Here, among others, uncertainty with its base in multicollinearity, outliers and heavy-tailed distributions were tackled. A Shapley value regression was applied to account for the multicollinearity in the data. The method provided very certain parameter estimates, but performed worse on prediction. As a result, the method can be reasonable in contexts where parameter certainty is more important than the predictive performance.

A robust MM-estimate was applied to counteract problems arising the latter two problems. The model's only observed benefit is a smaller MAE when outliers are present, although it performed similarly to an ordinary least squares in most cases. The method could still be useful for data with more outliers or a heavy-tailed error distribution. As a result, a residual analysis or an evaluation whether a lot of outliers are present is suggested in future works in order to identify reasons for the use of robust methods when performing modelling on other data sets. However, the model does not seem significantly beneficial for the data sets tested.

Modelling time-dependencies present in the model can prove useful. Although not shown for the real-world data, the simulations and theory indicate that time-dependencies should be incorporated if they exist. However, while incorporating time-dependencies between the data points, in the form of ARMA-process errors, can improve the quality of fit, it does not necessarily increase the certainty of the parameter estimates.

The linear Bayesian Structural Time Series (BSTS) model obtains the best quality of fit, but the certainty of the parameter estimates are low. It is clear that it is the model of choice if quality of fit is the only criteria. It is not determined why the other functional forms of BSTS does not perform as well as the linear, although suggestions have been made.

Incorporating prior beliefs into a model can be done in many different ways and here two have been tested. First, the constrained model constrains the optimisation space to only include acceptable options. This shows some promise but requires such solutions on the edge of this space to be acceptable. For example, by restricting values to be non-negative, a solution with parameters set to 0 has to be acceptable. Solving the regression problem using non-linear least-squares methods such as Gauss-Newton proved less interesting when multicollinearity exists, and another optimisation method should be used in these cases. This can be quickly

investigated through VIF-values, or investigating the singular value decomposition of the data matrix $\mathbf{X}$.

Second, the use of priors were tested in the Bayesian hierarchical model to nudge the solution in the right direction. The hierarchical model itself proved useful for predictive performance and parameter certainty, but the priors had varied results. Further, the hierarchical model is also harder and more bothersome to interpret, compared to many of the country-level models, due to the many smaller models which are part of the larger.

The results show varying outcomes for the use of the implemented non-parametric method XGBoost. The simulated data indicates that the model could potentially be used for the purpose of ROI estimation. The real data does however not show any evidence that the non-parametric model, as implemented, can be interpreted in a reliable sense. The bootstrap confidence intervals for the ROI estimates are very wide, indicating that the exact method used is not suitable for this purpose. However, the results of the datasets show in conjunction that it might be possible to use the method using SHAP values to obtain the ROI estimates from a non-parametric model, given it is less overfitted.

The different models almost all have their strengths and weaknesses. No model stand out as the best performer overall, however, the Shapley value regression, Bayesian hierarchical model and the linear Bayesian Structural Time Series have all been found to be strong candidates depending on the purpose.

6.1 Future Work

The area of MMM is a large area with many possibilities to explore. First and foremost, further development in interpretation of complex models, and obtaining ROI estimates in a theoretically sound, general way could be very interesting in an MMM context. One idea is to extend the theoretical foundation for the way to calculate the ROI-estimates using SHAP-values. The SHAP values shows promise since they are model-free and have theoretical guarantees, and could serve as the base for a universal method of interpretation. Given such a method more elaborated than the one in this study, further research could use a wide range of models that are hard to interpret in an MMM context and therefore also further research in the use of other non-parametric models within MMM.

Further, studies exploring causal relations and research in modelling of this causality is crucial due to the causal nature of MMM. To use regression in a MMM context can be strongly critiqued due to the non-causal relationships between the response variable and predictors.

Other interesting research areas are those not covered by, although incorporated into, this study. A few examples are ways in which to include seasonality and trend as well as variable selection into an MMM model. The way seasonality and trend is accounted for in this study should by all means be criticised, and other methods should be explored. The variable selection is a difficult issue due to and closely tying into causality. By maximising predictive performance, important factors could be left out. For example, if the temperature has an effect but is collinear with a media-spend variable leaving it out could improve predictive performance but overestimate the effect of the media variable. Research into reliable selection, which takes such effect into account could prove beneficial in an MMM context.

Lastly, elaborating methods combining the advantages of the different well-performing models could prove beneficial. Constructing a GLS model weighting the coefficients through a method similar to SVR or a hierarchical SVR model can prove difficult, although not infeasible.

Bibliography

[1] H. Akaike. "A new look at the statistical model identification". In: *IEEE Transactions on Automatic Control* 19.6 (Dec. 1974), pp. 716–723. ISSN: 0018-9286. DOI: 10.1109/TAC.1974.1100705.

[2] Sylvain Arlot and Alain Celisse. "A Survey of Cross Validation Procedures for Model Selection". In: *Statistics Surveys* 4 (July 2009). DOI: 10.1214/09-SS054.

[3] David Barber. *Bayesian Reasoning and Machine Learning*. New York, NY, USA: Cambridge University Press, 2012.

[4] James Bergstra and Yoshua Bengio. "Random Search for Hyper-parameter Optimization". In: *J. Mach. Learn. Res.* 13 (Feb. 2012), pp. 281–305. ISSN: 1532-4435. URL: http://dl.acm.org/citation.cfm?id=2188385.2188395.

[5] D Bertsimas, C Pawlowski, and Y.D. Zhuo. "From predictive methods to missing data imputation: An optimization approach". In: *Journal of Machine Learning Research* 18 (Apr. 2018), pp. 1–39.

[6] Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Berlin, Heidelberg: Springer-Verlag, 2006. ISBN: 0387310738.

[7] *Borden, N.H. (1964), The concept of the marketing mix, Journal of Advertising Research, Vol.4, pp. 2-7.* URL: http://books.google.co.uk/books?hl=en&lr=&id=9lmR75vPpEAC&oi=fnd&pg=PT15&dq=borden+and+the+concept+of+the+marketing+mix&ots=yWdRcn1XO-&sig=bpw7MjAy7kJ8Z1hVJFQ-Vd9Xa1Y.

[8] William Boulding, Eunkyu Lee, and Richard Staelin. "Mastering the Mix: Do Advertising, Promotion, and Sales Force Activities Lead to Differentiation?" In: *Journal of Marketing Research* 31.2 (1994), pp. 159–172. ISSN: 00222437. URL: http://www.jstor.org/stable/3152191.

[9] George E P Box and Norman R Draper. *Empirical Model-building and Response Surface*. New York, NY, USA: John Wiley & Sons, Inc., 1986. ISBN: 0-471-81033-9.

[10] G.E.P. Box, G.M. Jenkins, G.C. Reinsel, and G.M. Ljung. *Time Series Analysis: Forecasting and Control*. Wiley Series in Probability and Statistics. Wiley, 2015. ISBN: 9781118674925. URL: https://books.google.se/books?id=rNt5CgAAQBAJ.

[11] Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. New York, NY, USA: Cambridge University Press, 2004. ISBN: 0521833787.

[12] P.J. Brockwell, R.A. Davis, S.E. Fienberg, J.O. Berger, J. Gani, K. Krickeberg, I. Olkin, and B. Singer. *Time Series: Theory and Methods*. Springer Series in Statistics. Springer New York, 1991. ISBN: 9780387974293. URL: https://books.google.se/books?id=ZW%5C_ThhYQiXIC.

[13] Peter Bühlmann. "Bootstraps for Time Series". In: *Statist. Sci.* 17.1 (May 2002), pp. 52–72. DOI: 10.1214/ss/1023798998. URL: https://doi.org/10.1214/ss/1023798998.

[14] Peter Bühlmann. "Sieve bootstrap for time series". In: *Bernoulli* 3.2 (June 1997), pp. 123–148. URL: https://projecteuclid.org:443/euclid.bj/1177526726.

[15] Peter Lukas Bühlmann. "The blockwise bootstrap in time series and empirical processes". en. Diss. Math. Wiss ETH Zürich, Nr. 10354, 1993. Ref.: H. R. Künsch ; Korref.: E. Bolthausen. PhD thesis. ETH Zurich, 1993. DOI: 10.3929/ethz-a-000922255.

[16] Peter Bühlmann and Hans R Künsch. "Block length selection in the bootstrap for time series". In: *Computational Statistics & Data Analysis* 31.3 (1999), pp. 295–310. ISSN: 0167-9473. DOI: https://doi.org/10.1016/S0167-9473(99)00014-6. URL: http://www.sciencedirect.com/science/article/pii/S0167947399000146.

[17] Nicolas Carnot, Vincent Koen, and Bruno Tissot. *Economic Forecasting*. Palgrave Macmillan UK, 2005. ISBN: 9780230005815.

[18] G. Casella and R.L. Berger. *Statistical Inference*. Duxbury advanced series in statistics and decision sciences. Thomson Learning, 2002. ISBN: 9780534243128.

[19] David Chan and Mike Perry. *Challenges and Opportunities in Media Mix Modeling*. Tech. rep. Google, 2017.

[20] Tianqi Chen and Carlos Guestrin. "XGBoost: A Scalable Tree Boosting System". In: *Proceedings of the 22Nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. KDD '16. San Francisco, California, USA: ACM, 2016, pp. 785–794. ISBN: 978-1-4503-4232-2. DOI: 10.1145/2939672.2939785. URL: http://doi.acm.org/10.1145/2939672.2939785.

[21] E. Ward Cheney and David R. Kincaid. *Numerical Mathematics and Computing*. 6th. Pacific Grove, CA, USA: Brooks/Cole Publishing Co., 2007. ISBN: 0495114758.

[22] Per Christian Hansen, Victor Pereyra, and Godela Scherer. "Least Squares Data Fitting with Applications". In: *Least Squares Data Fitting with Applications* (Jan. 2012).

[23] Robert B. Cleveland, William S. Cleveland, Jean E. McRae, and Irma Terpenning. "STL: A Seasonal-Trend Decomposition Procedure Based on Loess (with Discussion)". In: *Journal of Official Statistics* 6 (1990), pp. 3–73.

[24] William S. Cleveland. "Robust Locally Weighted Regression and Smoothing Scatterplots". In: *Journal of the American Statistical Association* 74.368 (1979), pp. 829–836. ISSN: 01621459. URL: http://www.jstor.org/stable/2286407.

[25] John Crotts and Michael Wolfe. "Marketing Mix Modeling for the Tourism Industry: A Best Practices Approach". In: *International Journal of Tourism Sciences* 11 (Jan. 2011), pp. 1–15. DOI: 10.1080/15980634.2011.11434633.

[26] Jonathan D. Cryer and K.-S Chan. *Time Series Analysis: With Applications in R*. Jan. 2008, p. 491. DOI: 10.1007/978-0-387-75959-3.

[27] Saurabh Bagchi Deepen Garg Sagar Shah Vaswati Ghosh. *Multiplicative Marketing Mix Modeling simplified*. Tech. rep. 2013.

[28] Marnik G. Dekimpe and Dominique M. Hanssens. "The Persistence of Marketing Effects on Sales". In: *Marketing Science* 14.1 (Feb. 1995), pp. 1–21. ISSN: 1526-548X. DOI: 10.1287/mksc.14.1.1. URL: http://dx.doi.org/10.1287/mksc.14.1.1.

[29] Thomas J. Diciccio and Joseph P. Romano. "A Review of Bootstrap Confidence Intervals". In: *Journal of the Royal Statistical Society. Series B (Methodological)* 50.3 (1988), pp. 338–354. ISSN: 00359246. URL: http://www.jstor.org/stable/2345699.

[30] B. Efron. "Bootstrap Methods: Another Look at the Jackknife". In: *Ann. Statist.* 7.1 (Jan. 1979), pp. 1–26. DOI: 10.1214/aos/1176344552. URL: https://doi.org/10.1214/aos/1176344552.

[31] B. Efron and R. Tibshirani. "Bootstrap Methods for Standard Errors, Confidence Intervals, and Other Measures of Statistical Accuracy". In: *Statist. Sci.* 1.1 (Feb. 1986), pp. 54–75. DOI: 10.1214/ss/1177013815. URL: https://doi.org/10.1214/ss/1177013815.

[32] B. Efron and R.J. Tibshirani. *An Introduction to the Bootstrap*. Chapman & Hall/CRC Monographs on Statistics & Applied Probability. Taylor & Francis, 1994. ISBN: 9780412042317. URL: https://books.google.se/books?id=gLlpIUxRntoC.

[33] Bradley Efron. "Better Bootstrap Confidence Intervals". In: *Journal of the American Statistical Association* 82.397 (1987), pp. 171–185. ISSN: 01621459. URL: http://www.jstor.org/stable/2289144.

[34] Bradley Efron. "Nonparametric Standard Errors and Confidence Intervals". In: *The Canadian Journal of Statistics / La Revue Canadienne de Statistique* 9.2 (1981), pp. 139–158. ISSN: 03195724. URL: http://www.jstor.org/stable/3314608.

[35] Bradley Efron and Trevor Hastie. *Computer Age Statistical Inference: Algorithms, Evidence, and Data Science*. Institute of Mathematical Statistics Monographs. Cambridge University Press, 2016. DOI: 10.1017/CBO9781316576533.

[36] Bernd Fitzenberger. "The moving blocks bootstrap and robust inference for linear least squares and quantile regressions". In: *Journal of Econometrics* 82.2 (1998), pp. 235–287. ISSN: 0304-4076. DOI: https://doi.org/10.1016/S0304-4076(97)00058-4. URL: http://www.sciencedirect.com/science/article/pii/S0304407697000584.

[37] R.J. Freund, W.J. Wilson, and P. Sa. *Regression Analysis: Statistical Modeling of a Response Variable*. Elsevier Academic Press, 2006. ISBN: 9780120885978. URL: https://books.google.se/books?id=Qtx2lAEACAAJ.

[38] A. Gelman, J.B. Carlin, H.S. Stern, D.B. Dunson, A. Vehtari, and D.B. Rubin. *Bayesian Data Analysis, Third Edition*. Chapman & Hall/CRC Texts in Statistical Science. Taylor & Francis, 2013. ISBN: 9781439840955. URL: https://books.google.se/books?id=ZXL6AQAAQBAJ.

[39] F. Götze and H. R. Künsch. "Second-order correctness of the blockwise bootstrap for stationary observations". In: *Ann. Statist.* 24.5 (Oct. 1996), pp. 1914–1933. DOI: 10.1214/aos/1069362303. URL: https://doi.org/10.1214/aos/1069362303.

[40] Leigh J Halliwell. "The Gauss-Markov Theorem: Beyond the BLUE". In: ().

[41] Dominique M. Hanssens, Leonard J. Parsons, and Randall L. Schultz. *Response Models in Marketing*. Dordrecht: Springer Netherlands, 1990, pp. 100–130. ISBN: 978-94-009-1073-7. DOI: 10.1007/978-94-009-1073-7_1. URL: https://doi.org/10.1007/978-94-009-1073-7_1.

[42] Y.N. Harari. *Sapiens: A Brief History of Humankind*. Harper, 2015. ISBN: 9780062316103. URL: https://books.google.se/books?id=FmyBAwAAQBAJ.

[43] Frank E. Harrell Jr., Kerry L. Lee, and Daniel B. Mark. "Multivariable prognostic models: Issues in developing models, evaluating assumptions and adequacy, and measuring and reducing errors". In: *Statistics in Medicine* 15.4 (1996), pp. 361–387. DOI: 10.1002/(SICI)1097-0258(19960229)15:4<361::AID-SIM168>3.0.CO;2-4. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/.

[44] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *Elements of statistical learning: data mining, inference and prediction*. 2nd ed. Springer, 2009. URL: https://web.stanford.edu/~hastie/ElemStatLearn/.

[45] Peter J. Huber. "Robust Regression: Asymptotics, Conjectures and Monte Carlo". In: *Ann. Statist.* 1.5 (Sept. 1973), pp. 799–821. DOI: 10.1214/aos/1176342503. URL: https://doi.org/10.1214/aos/1176342503.

[46] R.J. Hyndman and G. Athanasopoulos. *Forecasting: principles and practice*. OTexts, 2014. ISBN: 9780987507105.

[47] insightr. *How Random Forests improve simple Regression Trees?* 2017. URL: https://www.r-bloggers.com/how-random-forests-improve-simple-regression-trees/ (visited on 04/03/2019).

[48] Yuxue Jin, Yueqing Wang, Yunting Sun, David Chan, and Jim Koehler. *Bayesian Methods for Media Mix Modeling with Carryover and Shape Effects*. Tech. rep. Google Inc., 2017.

[49] Katherine E. Jocz and John A. Quelch. "An Exploration of Marketing's Impacts on Society: A Perspective Linked to Democracy". In: *Journal of Public Policy & Marketing* 27.2 (2008), pp. 202–206. DOI: 10.1509/jppm.27.2.202. eprint: https://doi.org/10.1509/jppm.27.2.202. URL: https://doi.org/10.1509/jppm.27.2.202.

[50] Hans R. Kunsch. "The Jackknife and the Bootstrap for General Stationary Observations". In: *Ann. Statist.* 17.3 (Sept. 1989), pp. 1217–1241. DOI: 10.1214/aos/1176347265. URL: https://doi.org/10.1214/aos/1176347265.

[51] Steven L. Scott and Hal R. Varian. "Predicting the Present with Bayesian Structural Time Series". In: *Int. J. of Mathematical Modelling and Numerical Optimisation* 5 (Jan. 2014), pp. 4–23. DOI: 10.1504/IJMMNO.2014.059942.

[52] Stan Lipovetsky and Michael Conklin. "Analysis of regression in game theory approach". In: *Applied Stochastic Models in Business and Industry* 17.4 (2001), pp. 319–330.

[53] Yong Liu, Jorge Laguna, Matt Wright, and Hua He. "Media mix modeling – A Monte Carlo simulation study". In: *Journal of Marketing Analytics* 2.3 (Sept. 2014), pp. 173–186. DOI: 10.1057/jma.2014.3. URL: https://doi.org/10.1057/jma.2014.3.

[54] Robert Lucas. "Econometric policy evaluation: A critique". In: *Carnegie-Rochester Conference Series on Public Policy* 1.1 (1976), pp. 19–46. URL: https://EconPapers.repec.org/RePEc:eee:crcspp:v:1:y:1976:i::p:19-46.

[55] Scott M. Lundberg, Gabriel G. Erion, and Su-In Lee. "Consistent Individualized Feature Attribution for Tree Ensembles". In: *CoRR* abs/1802.03888 (2018). arXiv: 1802.03888. URL: http://arxiv.org/abs/1802.03888.

[56] Scott M Lundberg and Su-In Lee. "A Unified Approach to Interpreting Model Predictions". In: *Advances in Neural Information Processing Systems 30*. Ed. by I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett. Curran Associates, Inc., 2017, pp. 4765–4774. URL: http://papers.nips.cc/paper/7062-a-unified-approach-to-interpreting-model-predictions.pdf.

[57] K.V. Mardia, J.T. Kent, and J.M. Bibby. *Multivariate analysis*. Probability and mathematical statistics. Academic Press, 1979. ISBN: 9780124712508. URL: https://books.google.se/books?id=bxjvAAAAMAAJ.

[58] Daniel McNeish. "On Using Bayesian Methods to Address Small Sample Problems". In: *Structural Equation Modeling: A Multidisciplinary Journal* 23.5 (2016), pp. 750–773. DOI: 10.1080/10705511.2016.1186549. eprint: https://doi.org/10.1080/10705511.2016.1186549. URL: https://doi.org/10.1080/10705511.2016.1186549.

[59] Ecaterina Mhitarean-Cuvsinov. "Marketing Mix Modelling from multiple regression perspective". MA thesis. KTH Royal institute of technology, 2017.

[60] Timothy P. Whorf, Martin Wahlen, Robert B. Bacastow, Stephen C Piper, Charles D. Keeling, Martin Heimann, and Harro A.J. Meijer. *Atmospheric CO2 and 13CO2 Exchange with the Terrestrial Biosphere and Oceans from 1978 to 2000: Observations and Carbon Cycle Implications*. Jan. 2005. DOI: 10.1007/0-387-27048-5_5.

[61] Judea Pearl. "Causal inference in statistics: An overview". In: *Statist. Surv.* 3 (2009), pp. 96–146. DOI: 10.1214/09-SS057. URL: https://doi.org/10.1214/09-SS057.

[62] Jonas Peters, Dominik Janzing, and Bernhard Schölkopf. *Elements of causal inference: Foundations and learning algorithms*. MIT Press, 2017.

[63] Richard E. Quandt. "Estimating the Effectiveness of Advertising: Some Pitfalls in Econometric Methods". In: *Journal of Marketing Research* 1.2 (1964), pp. 51–60. ISSN: 00222437. URL: http://www.jstor.org/stable/3149922.

[64] J.O. Rawlings, S.G. Pantula, and D.A. Dickey. *Applied Regression Analysis: A Research Tool*. Springer Texts in Statistics. Springer New York, 2001. ISBN: 9780387984544. URL: https://books.google.se/books?id=PMeJGeXA09EC.

[65] Joseph P. Romano and Michael Wolf. "Improved nonparametric confidence intervals in time series regressions". In: *Journal of Nonparametric Statistics* 18.2 (2006), pp. 199–214. DOI: 10.1080/10485250600687812. eprint: https://doi.org/10.1080/10485250600687812. URL: https://doi.org/10.1080/10485250600687812.

[66] P.E. Rossi, G.M. Allenby, and R. McCulloch. *Bayesian Statistics and Marketing*. Wiley Series in Probability and Statistics. Wiley, 2012. ISBN: 9780470863688. URL: https://books.google.se/books?id=GL8VS9i%5C_B2AC.

[67] R.H. Shumway and D.S. Stoffer. *Time Series Analysis and Its Applications: With R Examples*. Springer Texts in Statistics. Springer International Publishing, 2017. ISBN: 9783319524511. URL: https://books.google.se/books?id=PTkoMQAACAAJ.

[68] Andrej-Nikolai Spiess and Natalie Neumeyer. "An evaluation of R2 as an inadequate measure for nonlinear models in pharmacological and biochemical research: a Monte Carlo approach." eng. In: *BMC Pharmacol* 10 (2010), p. 6. ISSN: 1471-2210 (Electronic); 1471-2210 (Linking). DOI: 10.1186/1471-2210-10-6.

[69] Yunting Sun, Yueqing Wang, Yuxue Jin, David Chan, and Jim Koehler. *Geo-level Bayesian Hierarchical Media Mix Modeling*. Tech. rep. 2017.

[70] Gerard J. Tellis. "Modeling Marketing Mix". In: 2006.

[71] J.W. Tukey. *Exploratory Data Analysis*. Addison-Wesley series in behavioral science. Addison-Wesley Publishing Company, 1977. ISBN: 9780201076165. URL: https://books.google.se/books?id=UT9dAAAAIAAJ.

[72] Rahmi Wahidah Siregar, Tulus Tulus, and Marwan Ramli. "Analysis Local Convergence of Gauss-Newton Method". In: *IOP Conference Series: Materials Science and Engineering* 300 (Jan. 2018), p. 012044. DOI: 10.1088/1757-899X/300/1/012044.

[73] Larry Wasserman. *All of Statistics: A Concise Course in Statistical Inference*. Springer Publishing Company, Incorporated, 2004.

[74] Johan Wejryd. "On Consumed Democracy?: The Expansion of Consumer Choice, Its Causal Effects on Political Engagement, and Its Implications for Democracy". In: (2018).

[75] Richard Williams. *Lecture notes in Graduate Statistics II, Lecture 11*. Jan. 2015. URL: https://www3.nd.edu/~rwilliam/stats2/l11.pdf.

[76] Jingtao Yao, Nicholas Teng, Hean-Lee Poh, and Chew Lim Tan. "Forecasting and Analysis of Marketing Data Using Neural Networks". In: *J. Inf. Sci. Eng.* 14 (1998), pp. 843–862.

[77] Victor J. Yohai. “High Breakdown-Point and High Efficiency Robust Estimates for Regression”. In: *Ann. Statist.* 15.2 (June 1987), pp. 642–656. DOI: 10.1214/aos/1176350366. URL: https://doi.org/10.1214/aos/1176350366.

[78] Chun Yu and Weixin Yao. “Robust linear regression: A review and comparison”. In: *Communications in Statistics - Simulation and Computation* 46.8 (2017), pp. 6261–6282. DOI: 10.1080/03610918.2016.1202271. eprint: https://doi.org/10.1080/03610918.2016.1202271. URL: https://doi.org/10.1080/03610918.2016.1202271.

A Complementary theory

A.1 Maximum likelihood for continuous predictors

The maximum likelihood estimation is not as trivial in the continuous case and instead the formulation

$$\frac{f(y|\theta_1)}{f(y|\theta_2)} \tag{A.1}$$

is used to compare two parameter values. This formulation is stated by assuming that the distribution is differentiable on the interval. It is then possible to use the mean value theorem to get the result $P(y-\epsilon < Y < y+\epsilon|\theta) = 2\epsilon f(\xi|\theta)$ with $\xi \in [y-\epsilon, y+\epsilon]$. With this result, the comparison can be stated as:

$$\frac{P(y-\epsilon < Y < y+\epsilon|\theta_1)}{P(y-\epsilon < Y < y+\epsilon|\theta_2)} = \frac{2\epsilon f(\xi|\theta_1)}{2\epsilon f(\xi|\theta_2)} = \frac{f(\xi|\theta_1)}{f(\xi|\theta_2)} \tag{A.2}$$

By then letting $\epsilon \to 0$, ξ will converge to the realized value y and the formulation then equals the former. From this formulation, a value larger than 1 suggests that θ_1 is more likely than θ_2. As a result the maximum likelihood estimator is the most likely estimator since

$$\frac{f(\xi|\theta^{MLE})}{f(\xi|\theta')} \geq 1$$

A.2 Maximum Likelihood for linear regression

Under the assumptions stated in section 2.2.1 the likelihood function for linear regression is stated as:

$$\begin{aligned}
\mathcal{L}(\beta; X, y) &= f(y|X, \beta) \\
&= \mathcal{N}(X\beta, \sigma^2 \mathbb{I}_n) \\
&= \prod_{i=1}^{n} \mathcal{N}(X_i^T \beta, \sigma^2) \\
&= \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma^2}} \exp(-\frac{(y_i - x_i^T \beta)^2}{2\sigma^2})
\end{aligned}$$

The MLE can then be found through the maximisation of the log-likelihood, resulting in:

$$\begin{aligned}
\hat{\beta}^{mle} &= \arg\max_{\beta} \log(\mathcal{L}(\beta; X, y)) \\
&= \arg\max_{\beta} -\sum_{i=1}^{n} \Big(\frac{(y_i - x_i^T\beta)^2}{2\sigma^2} - \underbrace{\log(\sqrt{2\pi\sigma^2})}_{=c_i} \Big) \\
&= \arg\min_{\beta} \frac{1}{2\sigma^2} \sum_{i=1}^{n} (y_i - x_i^T\beta)^2 - \underbrace{\sum_{i=1}^{n} c_i}_{=C} \\
&= \arg\min_{\beta} \sum_{i=1}^{n} (y_i - x_i^T\beta)^2 = \hat{\beta}^{ols}
\end{aligned}$$

As can be seen above, the optimization for the MLE and the OLS estimator can be stated as the same problem. The maximum likelihood estimator is therefore equivalent and can be used interchangeably to the least squares estimator under these assumptions.

A.3 Forecasting MA-processes

An $MA(q)$-process is not as clear-cut to forecast as an $AR(p)$-process since there are no realisations of the errors available. They are not trivial to estimate either since estimation such as $\hat{e}_t = x_t - \hat{x}_t$ would need errors to estimate x_t. However, the property of invertability can be used instead. An $MA(q)$-process is invertible if the roots of the characteristic polynomial lie outside the unit-circle [10]. Further, an invertible $MA(q)$-process can be stated as an $AR(\infty)$-process [12]; which solves this issue. The error can therefore be estimated as a combination of the previous observations if the restriction to invertible $MA(q)$-processes is made. Using an $MA(1)$-process as an example, $X_t = \theta e_{t-1} + e_t$ with $|\theta| < 1$, the errors can be written as:

$$\begin{aligned}
e_t &= X_t - \theta e_{t-1} \\
&= X_t - \theta X_{t-1} + \theta^2 e_{t-2} \\
&\vdots \\
&= \sum_{j=0}^{\infty} (-\theta)^j X_t
\end{aligned}$$

Since a realisation of a process does not contain an infinite amount of measurements, this estimation will have to be cut of. This will, however, still be reasonable for errors late in the series as the weights decreases exponentially with time. The expected error of an $MA(1)$-process with a realisation $x_t : t = 1, .., n$ therefore reads:

$$\begin{aligned}
\mathbb{E}[e_t | X_{1:t} = x_{1:t}] &= \mathbb{E}[\sum_{j=0}^{\infty} (-\theta)^j X_{t-j} | X_{1:t} = x_{1:t}] \\
&= \sum_{j=0}^{t-1} (-\theta)^j x_{t-j} + \sum_{j=t}^{\infty} (-\theta)^j \mathbb{E}[X_{t-j} | X_{1:t} = x_{1:t}] \\
&= \sum_{j=0}^{t-1} (-\theta)^j x_{t-j} + (-\theta)^t \mathbb{E}[e_0 | X_{1:t} = x_{1:t}]
\end{aligned}$$

In estimation of this quantity, two approaches are to either set the initial value to zero, $\hat{e}_0 = 0$, or us a technique called back-forecasting in order to estimate the conditional mean. The first method is simpler than the second and if t is large enough the differences between the methods will be negligible [10]. Thus, given the parameter θ and a sufficiently large series, we can estimate the errors as follows:

$$\hat{e}_t = \sum_{j=0}^{t} (-\theta)^j x_{t-j}. \tag{A.3}$$

In turn this yields the lead 1 forecast $\hat{x}_{t+1}^{(1)} = \theta\hat{e}_t$. In order to make a lead 2 forecast, the error at lead 1 has to be estimated. However, under the iid assumption, the error term e_t at time-point t is independent from $X_k, k < t$ and we have that $\mathbb{E}[e_{t+1}|X_{1:t}] = 0$ [10]. This results in the expectation

$$[X_{t+2}|X_{1:t}] = [e_{t+2}|X_{1:t}] + \theta[e_{t+1}|X_{1:t}] = 0 \tag{A.4}$$

and therefore the best estimation, in the least squares sense, is $\hat{x}_{t+2}^{(2)} = 0$. These types of forecasts can be further generalised in order to incorporate higher order processes. The error estimation can be done in a similar manner for higher order processes by recursive estimation, cf. [12]. Taking the conditional expectation of x_{t+l} from an $MA(q)$-process yields the following expression:

$$\mathbb{E}[X_{t+1}|X_{1:t}] = \mathbb{E}[e_{t+l} + \theta_1 e_{t+l-1} + ... + \theta_q e_{t+l-q}|X_{1:t}] = \sum_{j=1}^{q} \theta_j \mathbb{E}[e_{t+l-j}|X_{1:t}]. \tag{A.5}$$

Plugging in the estimated errors and using independence of future errors, leads to the lead l forecast

$$\hat{x}_k^{(l)} = \sum_{i=l}^{q} \theta_i \hat{e}_{k-i} \tag{A.6}$$

of an $MA(q)$-process.

B Additional methodology information

Parameter name	True coefficient value	Included in modelling
Intercept	10	Yes
sunshine	0	Yes
precipitation	-0.005	Yes
temperature	0	No
competitor_a	-0.036	Yes
competitor_b	0	Yes
competitor_c	0	Yes
cpi	0	No
cci	0	Yes
gdp	0.019	Yes
product_a.x	-0.006	Yes
product_b.x	0	No
product_c.x	0	No
product_a.y	0.020	Yes
product_b.y	0.089	No
product_c.y	0.094	No
event_a	0.014	Yes
event_b	0.051	No
display...net	0.067	Yes
facebook...net	0.133	Yes
search_branded...net	0.167	Yes
display...impression	0	Yes
facebook...impression	0	No
search_branded...impression	0	No
display...cpm	0.049	No
facebook...cpm	0.089	No
search_branded...cpm	0.111	No
Christmas	0.333	Yes

Table B.1: The true values for the coefficients, and whether they were included for the models (except for BSTS, which had its own variable selection method).

	Simulated data	BSTS	Hierarchical models	XGBoost
Seed used	123	2016	42	123

Table B.2: The seeds used in the different procedures incorporating a random element.

C Additional results

C.1 Real data

C.1.1 Coefficient estimates

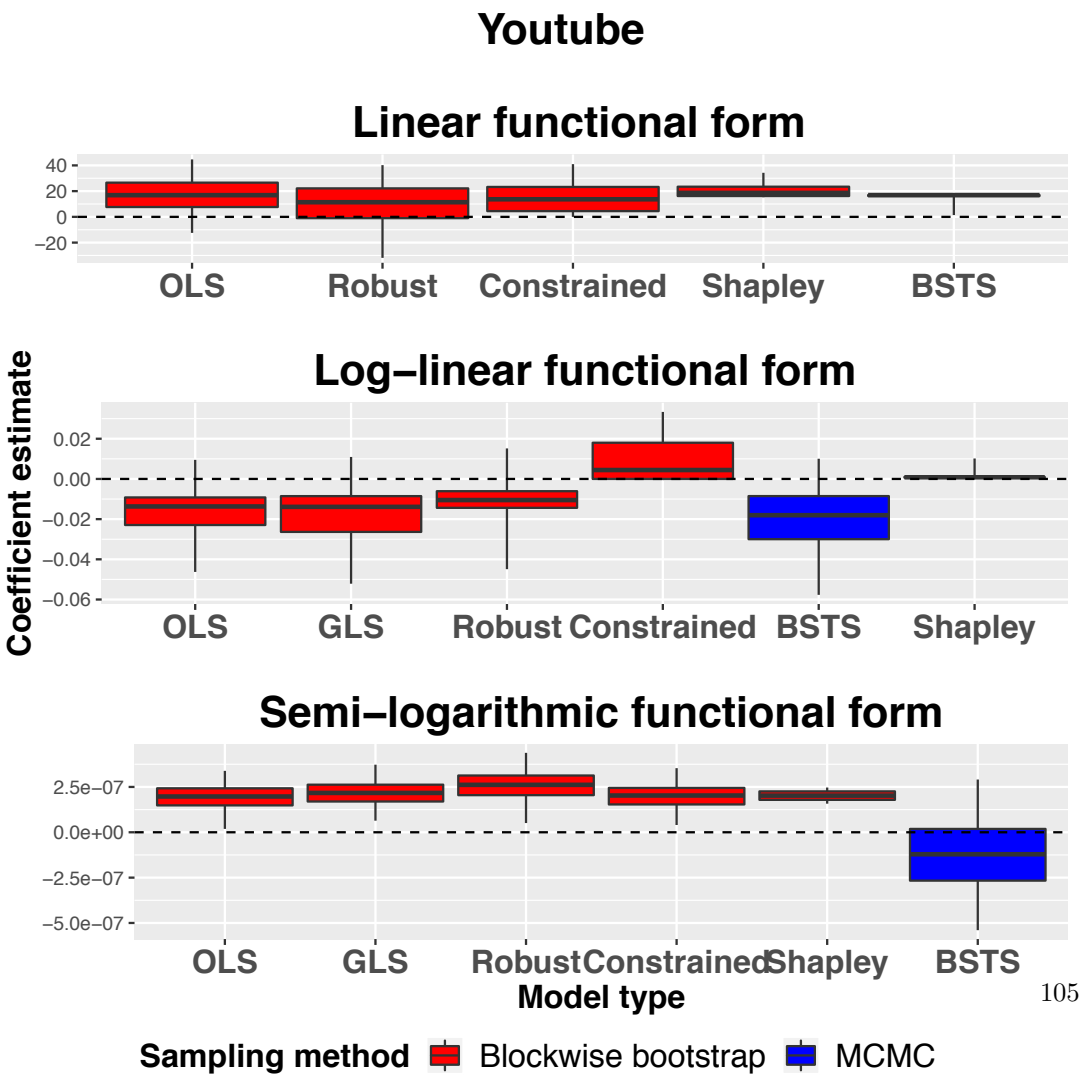


Figure C.1: Coefficient estimates for the Youtube predictor on the real data.

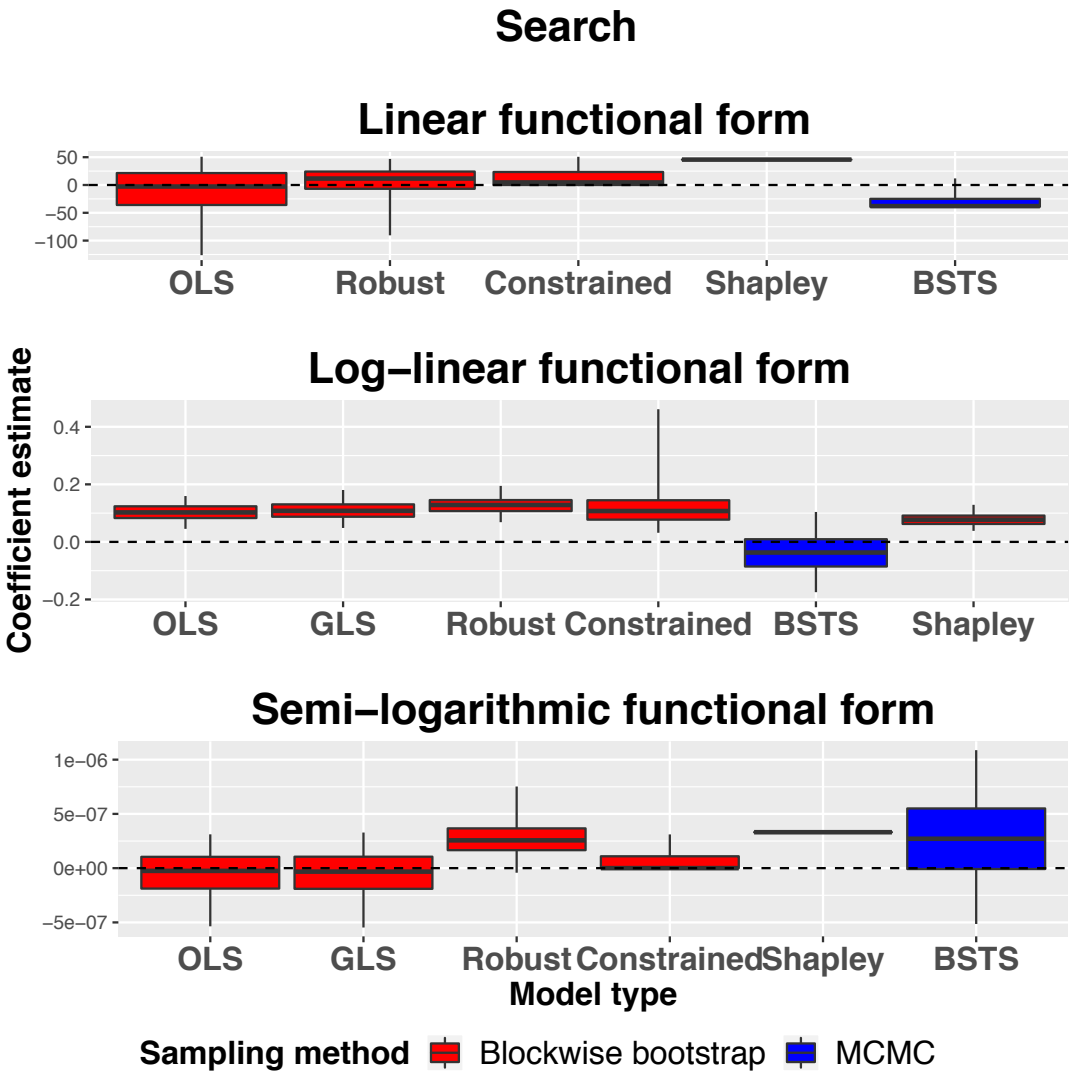


Figure C.2: Coefficient estimates for the Search predictor on the real data.

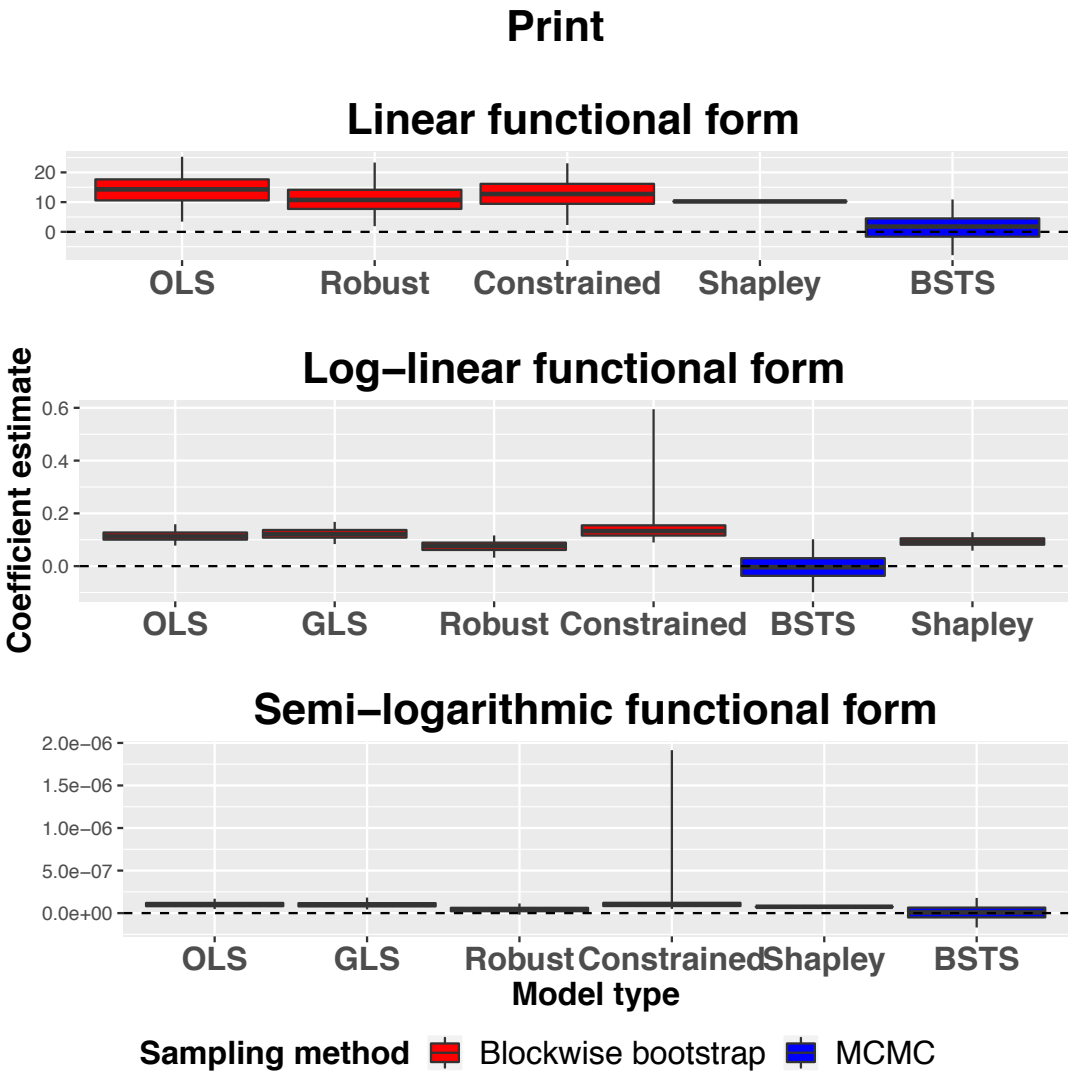


Figure C.3: Coefficient estimates for the Print predictor on the real data.

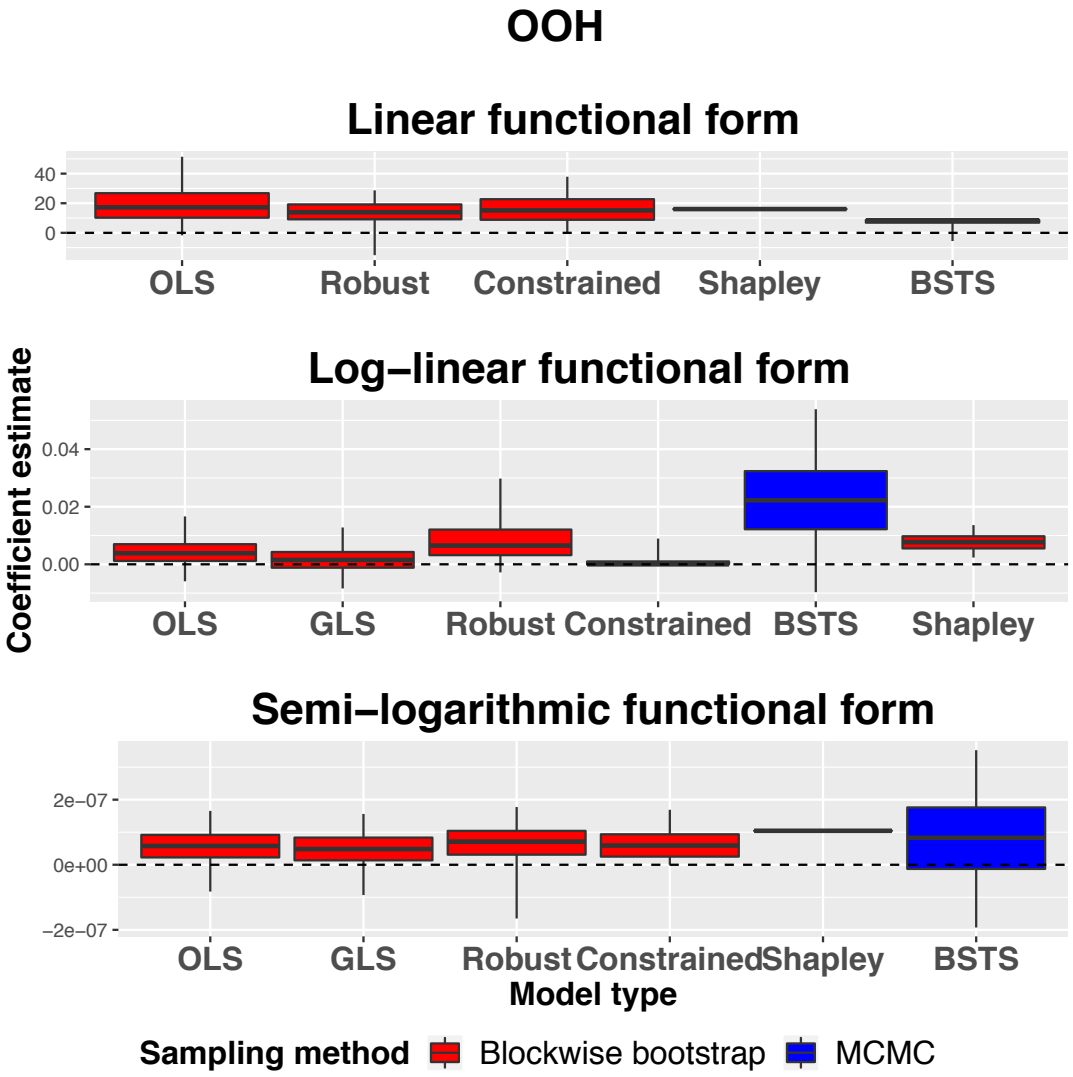


Figure C.4: Coefficient estimates for the OOH predictor on the real data.

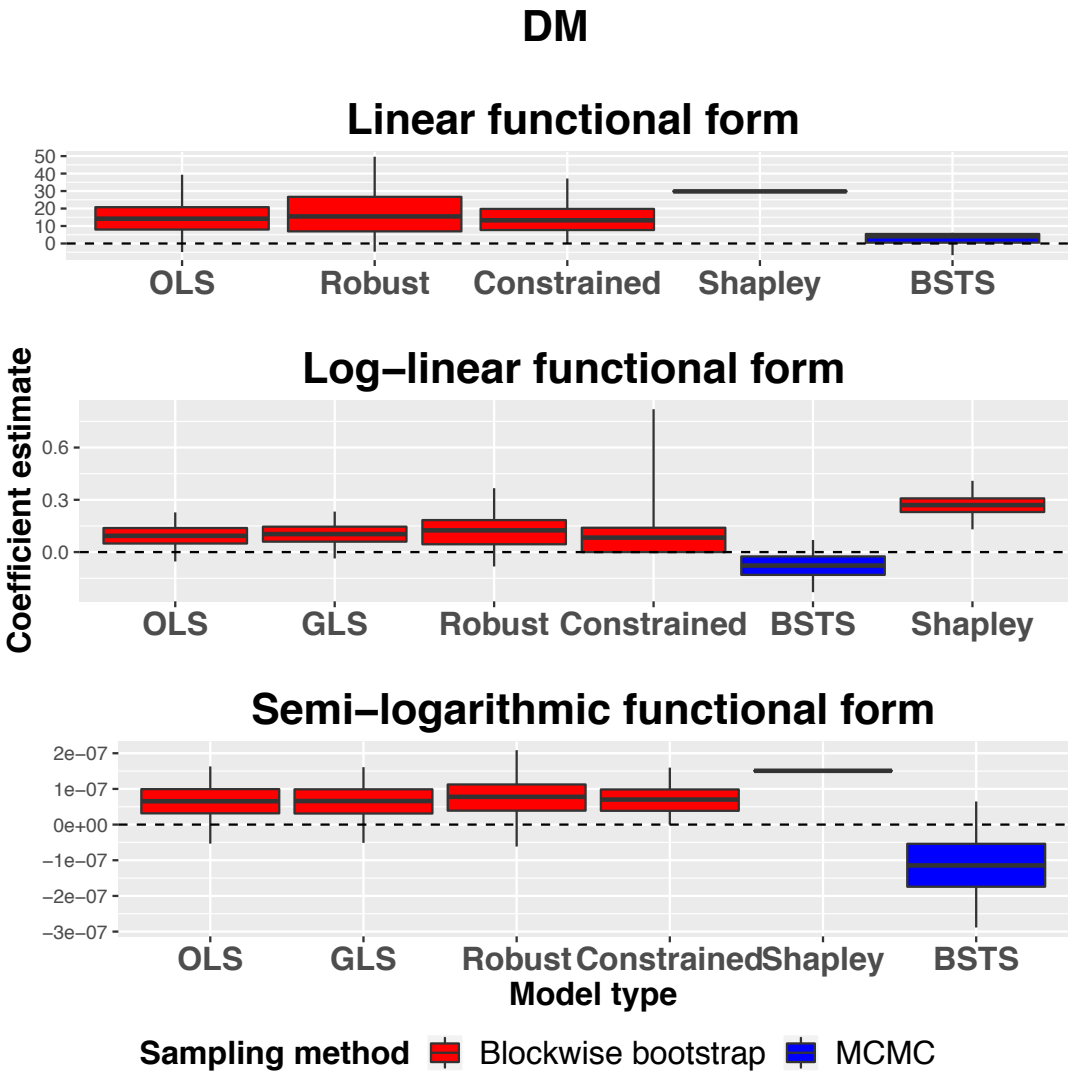


Figure C.5: Coefficient estimates for the DM predictor on the real data.

C.1.2 ROI estimates

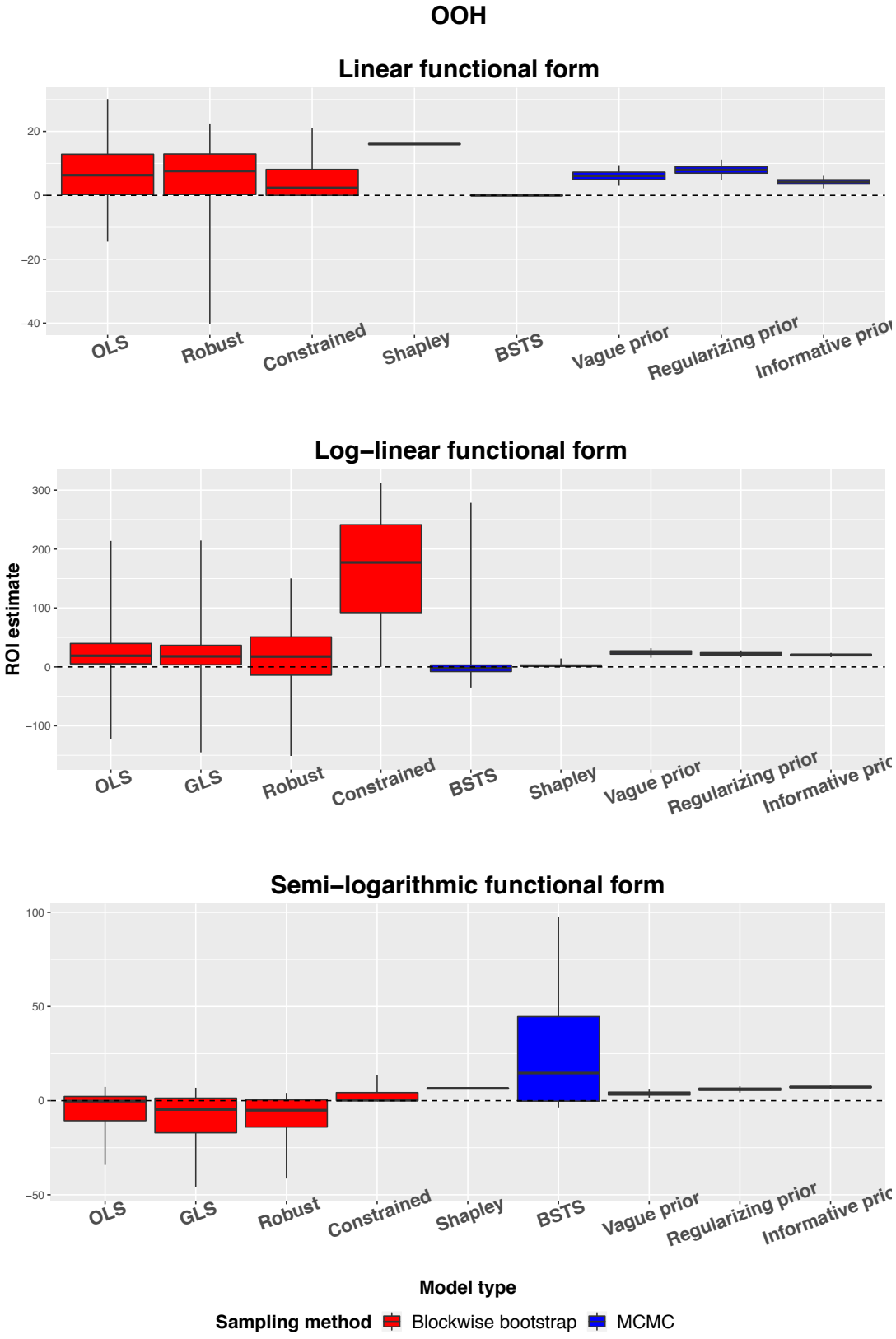


Figure C.6: ROI estimates for the OOH spend on the real data.

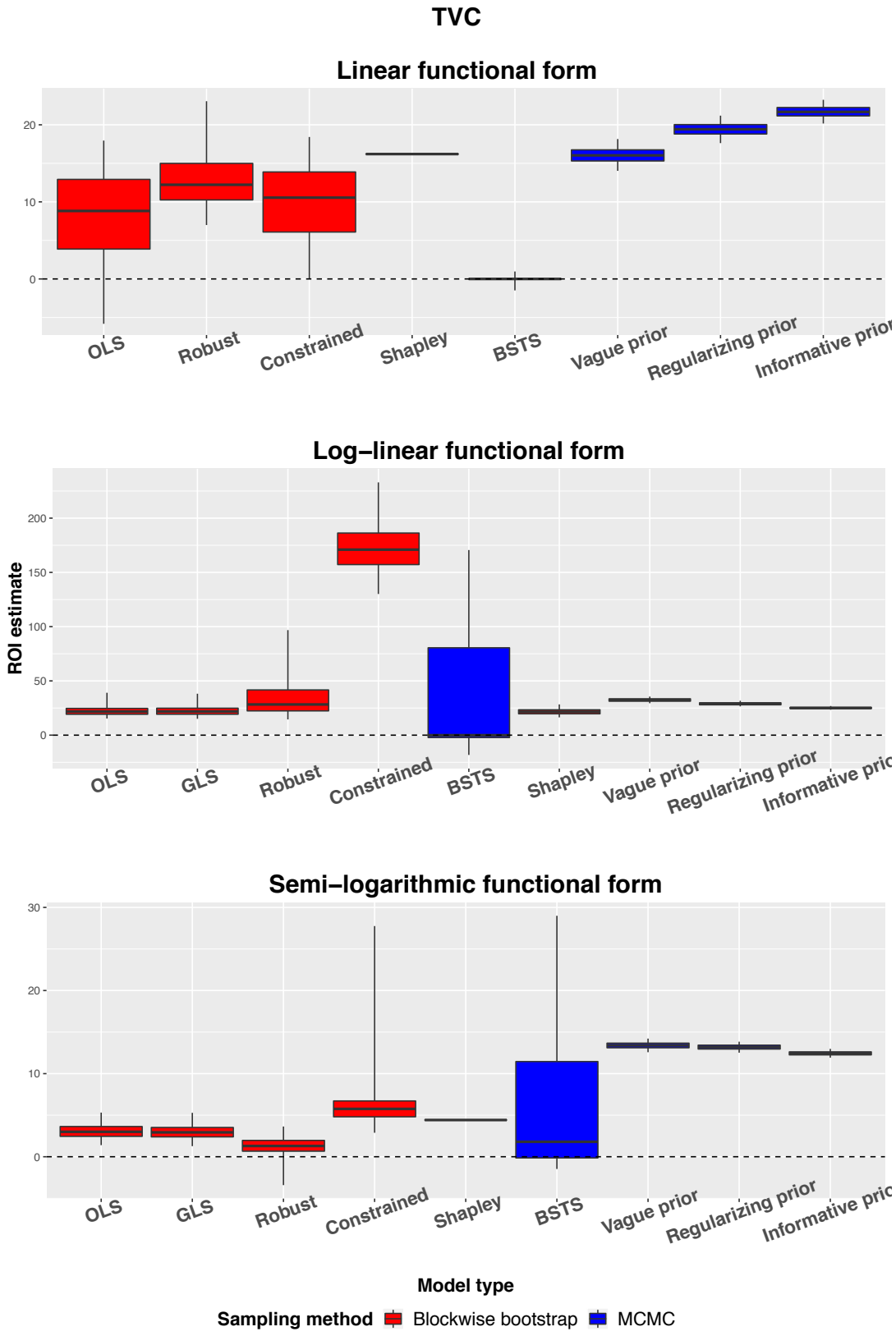


Figure C.7: ROI estimates for the TVC spend on the real data.

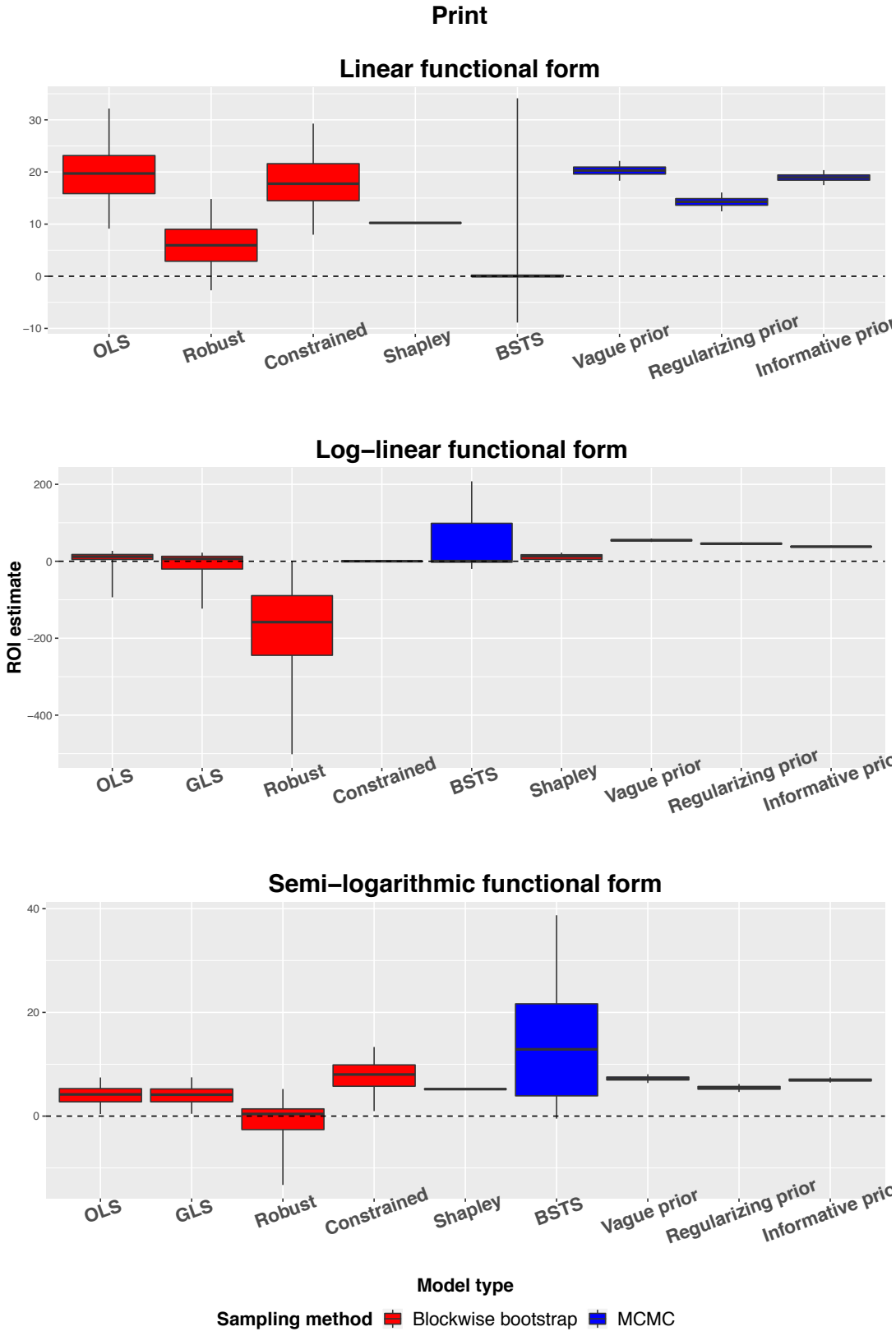


Figure C.8: ROI estimates for the Print spend on the real data.

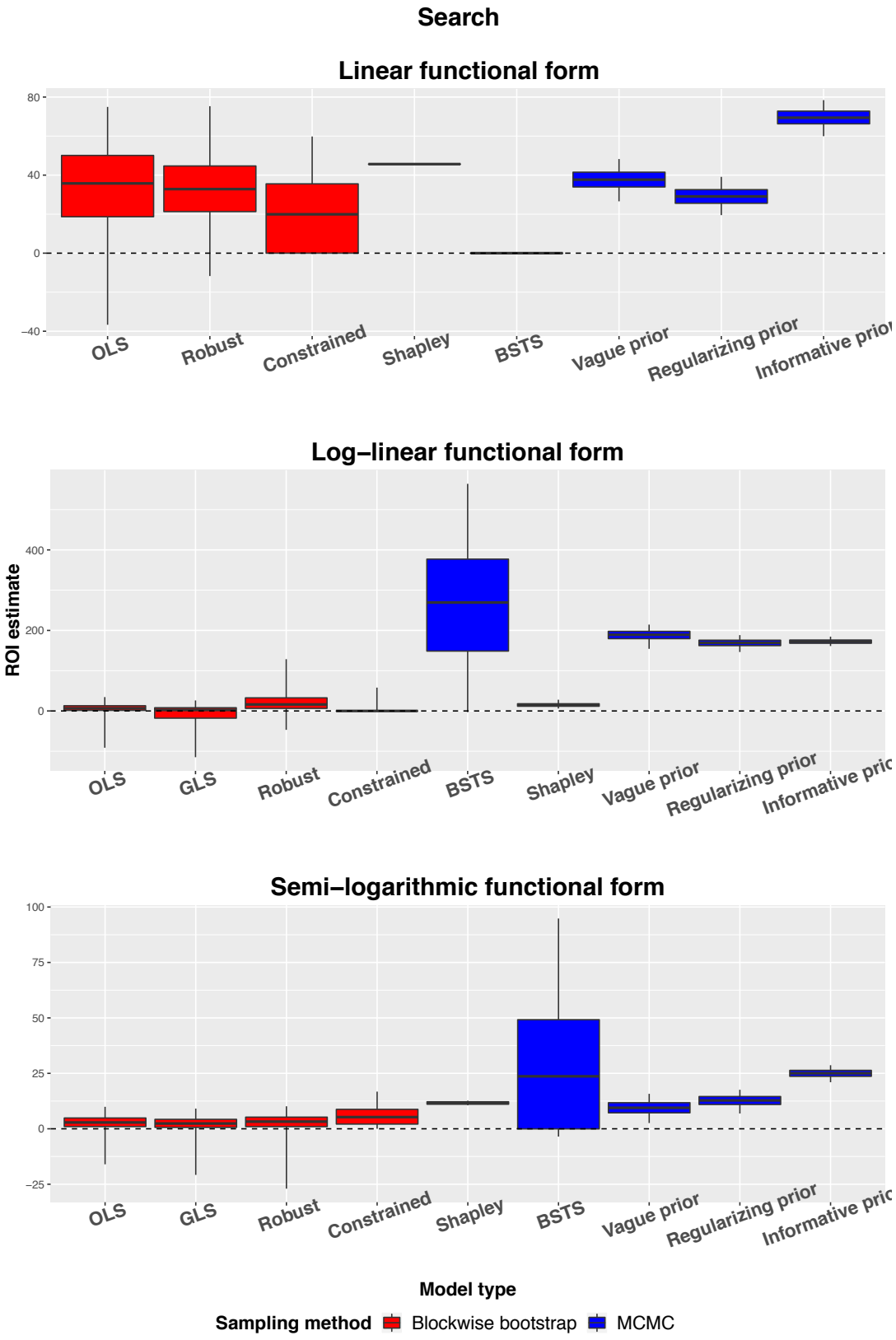


Figure C.9: ROI estimates for the Search spend on the real data.